Asymptotic behaviour, non-local dynamics and data assimilation tailoring of the reduced $\kappa-\varepsilon$ model to address turbulent transport of fusion plasmas

D. Auroux,[1] P. Ghendrih,[2] L. Lamerand,[1] F. Rapetti,[1] and E. Serre[3]
[1)] *Université Côte d'Azur, Inria, CNRS, LJAD, F-06108 Nice*
[2)] *IRFM, CEA Cadarache, F-13108 St. Paul-lez-Durance*
[3)] *M2P2, Aix Marseille Univ, CNRS, Centrale Marseille, F-13451 Marseille*

(*Electronic mail: eric.serre@univ-amu.fr)

(Dated: 5 September 2022)

The high-dimensional and multiscale nature of fusion plasma flows require the development of reduced models to be implemented in numerical codes capable of capturing the main features of turbulent transport in a sufficiently short time to be useful during tokamak operation. This paper goes further in the analysis of the dynamics of the $\kappa-\varepsilon$ model based on the turbulent kinetic energy κ and its dissipation rate ε [Baschetti *et al.*, Nuc. Fus 61, 106020 (2021)] to improve the predictability of the transverse turbulent transport in simulation codes. Present 1D results show further capabilities with respect to current models (based on constant effective perpendicular diffusion) and on the standard quasi-linear approach. The nonlinear dependence of D in κ and ε estimated from two additional transport equations allow to introduce some non-locality in the transport model. This is illustrated by the existence of parameter ranges with turbulence spreading. The paper also addresses another issue related to the uncertainties on the inherent free parameters of such reduced model. The study proposes a new approach in the fusion community based on a variational data assimilation involving the minimisation of a cost function defined as the distance between the reference data and the calculated values. The results are good, and show the ability of the data assimilation to reduce uncertainties on the free parameters, which remains a critical point to ensure the total reliability of such an approach.

I. INTRODUCTION

Fusion based on magnetic confinement aims at producing power by using the energy liberated by fusing deuterium and tritium nuclei at extremely high temperatures (10^7-10^8K) within a plasma confined by magnetic fields in machines of toroidal shape known as tokamaks. This is thus a promising alternative energy source that could potentially fulfill the high demand for electricity in the next future. The International Tokamak Experimental Reactor (ITER) has been designed as the key step between today's fusion research machines and tomorrow's fusion power plants. Regarding the expected thermonuclear plasma performance, ITER operation requires an unprecedented effort on the way to controlling plasma heat and particle fluxes[1], in which numerical simulations play a critical role to design optimal scenarios.

Despite the exponential growth of computer speed along with significant improvements in computer technology, the numerous physics and engineering issues to address as well as the very large number of degrees of freedom to handle require the development of a chain of models. This latter ranges from low to high fidelity models, from simplified models for optimization and uncertainty propagation to state-of-the-art first principle models of plasma turbulence transport in relevant plasma conditions. In this context, two-dimensional fluid transport codes (see exhaustive references in Refs.[2,3]) rely on models in which plasma turbulence has been smoothed out by averaging to provide relevant information on appropriate return times. It is similar to the Reynolds Averaged Navier–Stokes (RANS) codes commonly used for engineering applications in computational fluid dynamics (CFD)[4]. One of the main issues of these 2D transport codes is the accurate modeling of the transverse turbulent fluxes resulting from the averaging of stresses due to fluctuations. Although intermittency, long-range correlations and ballistic transport events have been widely documented in the plasma edge for a long time[5], all current transport codes in the community (see e.g.[2] and[3]) are still based, for reasons of simplicity and numerical efficiency, on the assumption that these transverse flows are driven by local gradients, i.e. by Fick's laws with constant effective perpendicular diffusion coefficients. These ad-hoc coefficients are indeed not determined self-consistently but are tuned by hand to match experimental data. These turbulent particles diffusion D_t, turbulent viscosity ν_t and turbulent heat conductivity ξ_t (simplicity here $D_t=\nu_t=\xi_t$) give rise to effective coefficients in the transport equations that must be determined at each point of the mesh. The issue here is that these ad-hoc coefficients being flow-dependent differ from one machine to another, from one pulse to another in the same device and even from one location to another in a given discharge[6]. They must then be considered as free parameters with an extremely large number of degrees of freedom, which reduces drastically the predictive capabilities of these codes.

To circumvent this limitation, the option of a direct coupling between a mean-field code and a local (flux-tube) plasma turbulence code was proposed by Hasenbeck *et al.*[7] for example. In such a scheme, the mean-field code still follows a gradient-diffusion hypothesis to describe transverse transport, but the transport coefficients are dynamically and self-consistently obtained from the output of turbulence simulations run in a gradient-driven manner (with local plasma parameters and gradients used as inputs) at well-chosen sampling positions in time and space. This approach has been proved successful at reproducing plasma profiles in simple

test cases. However, the application to complex cases remains to be undertaken. Additionally, a number of fundamental questions remain in this context, e.g. how to efficiently determine the sampling points and decisive parameters from the mean-field code for the local turbulence simulations, how their choice affects the stability, convergence and performance of such a coupled code system, and if the underlying hypothesis of a scale separation between macroscale (mean-field) and mesoscale (local turbulence) dynamics always holds in the regimes and regions of interest to ensure the feasibility of the approach. Also the physical limitations of this concept as compared to global turbulence simulations and the computational costs still have to be assessed in depth. Similar problems arise when local turbulence codes are used to build a database of transport coefficients depending on the mean-field code's input parameters[8]. To solve the computational cost issue, another approach consists in completing the model with equations describing the time and/or space evolution of the transport coefficients, thus making the call to a turbulence code unnecessary. This approach has already been used successfully in the frame of 1D models for L-H transition studies[9,10] and can potentially be applied to 2D or 3D edge mean-field models. The consequence is the substitution, as free parameters, of perpendicular diffusion coefficients with parameters defining key properties of the underlying transport mechanisms (e.g., turbulence growth and damping rates). Although this might not lead to a reduction of the number of free parameters, the new parameters are expected to be more driven by the underlying transport physics and hence will vary much less from one machine to the other or from one location to another in the same plasma. Also, their dependencies with plasma characteristics can in general be derived or constrained by theoretical considerations.

Very recently we proposed an innovative path for fusion plasma simulations inspired from computational fluid dynamics, namely from a $\kappa-\varepsilon$ like model developed for the Reynolds-Averaged Navier–Stokes approach designed in the 70's[11] but still largely used today to investigate many engineering flows. The model relies on to the evolution of the turbulence kinetic energy per unit mass of the fluctuating transverse velocity $\kappa = \frac{1}{2}\langle \tilde{v}^2 \rangle$. ε relates to a damping process acting on κ as the dissipation rate in neutral fluids to determine the turbulent diffusion with $D_t = C\kappa^2/\varepsilon$, where C is a proportionality coefficient and v is the drift velocity. Indeed, the scales of dissipation of the turbulent energy in plasma being still an open question[12], the usual definition as a rate of dissipation at the Kolmogorov scale is less meaningful in the context of plasma turbulence, so the field ε is simply understood as a damping process, or predator of the turbulent energy. The success of this kind of approaches relies on the fact that these constants are characteristic of fundamental processes of turbulence, and are hence less dependent on the plasma discharge in the same tokamak, in contrast to the initial diffusion coefficient D_t. Results in Baschetti *et al.*[13,14] showed the remarkable potential of this model to capture key aspects of the physics of turbulent transport throughout the plasma.

However, as any reduced model the transport equations for κ and ε induce few free parameters that have to be fixed. In current version[13,14], these parameters are fixed from theoretical considerations based on features of the local dynamics of the $\kappa-\varepsilon$ system and the known properties of transport as well as on a dimensional analysis in the frame of the Kolmogorov theory for homogeneous isotropic turbulence. All these considerations have however a limited validity, and the calibration of these free parameters from available experimental and/or micro-turbulence simulations data of reference would certainly still improve the predictive capability of the model. It is then mandatory to identify the model parameters that will make the computed solution of the model precisely correspond to the data of reference, and to investigate their impact on the dynamic of the system.

Typically used in geosciences, oceanography, or meteorology[15,16] where the atmospheric model must precisely match current and past measurements to obtain an accurate forecast, data assimilation allows to work on models with a very large number of parameters confronted to observations of different accuracy not homogeneously spread in space and time. The present work is based on a Variational Data Assimilation (VDA) which involves the minimization of a cost function defined as the distance between the data of reference and the computed values by the model[17]. A successful implementation of such procedure in current 2D fluid transport codes for fusion plasmas like SOLEDGE3X[3] or SOLPS-ITER[2] would allow a precise estimation of the remaining free parameters from the data of reference, or would provide an explicit trajectory for a parameter which could be compared to some considered closure laws.

In this paper, we go further in the analysis of the dynamical system composed by the two transport equations for κ and ε derived in[14] and reduced to 1D in the radial direction by averaging in the poloidal and toroidal directions (Section II). The data assimilation technique is presented in Section III. Based on an appropriate normalization, the 1D analysis allows to isolate the fundamental parameters that determine the local behavior of the dynamical system and then the properties of the solutions when varying them (Section IV). The potential of the variational data assimilation is finally illustrated in two spreading regimes (Section V). Concluding remarks end the paper (Section VI).

II. REDUCED MODEL

We consider here the κ-ε model reduced to 1-D in the radial direction following the procedure defined in[14] (Section 3). As mentioned in the Introduction, $\kappa = \frac{1}{2}\langle \tilde{v}^2 \rangle$ and ε relates a damping process acting on κ to determine the turbulent diffusion with $D_t = C\kappa^2/\varepsilon$, where C is constant. On contrary to Ref.[14], the model is here decoupled from fluid plasma equations and thus plasma flow variables will be considered as constant in time in the following.

A. Transport model for κ and ε

The evolution of the two fields κ-ε is governed by local dynamics and transverse and parallel transport. Both fields κ and ε are understood to be proportional to energies and thus defined to be positive. The radial variable r varies between 0 and a the little radius of the torus, corresponding to the core boundary of the plasma and the tokamak wall, respectively. The model equations and (initial, boundary) conditions are detailed in[14] and read:

$$\partial_\tau \kappa - \frac{1}{r}\nabla_r\left(rC_D^\kappa \frac{\kappa^2}{\varepsilon}\nabla_r\kappa\right) = \gamma_s^\kappa \kappa - \frac{1}{D_\omega}\kappa^2 - \varepsilon - \sigma_\parallel^\kappa \kappa, \quad (1a)$$

$$\partial_\tau \varepsilon - \frac{1}{r}\nabla_r\left(rC_D^\varepsilon \frac{\kappa^2}{\varepsilon}\nabla_r\varepsilon\right) = \gamma_s^\varepsilon \varepsilon - \frac{V}{\kappa^{3/2}}\varepsilon^2 - \sigma_\parallel^\varepsilon \varepsilon, \quad (1b)$$

$$\begin{aligned} &\kappa(\tau=0,r)=\kappa_0, \quad \nabla_r\kappa(\tau,r=0)=0, \quad \kappa(\tau,r=a)=0,\\ &\varepsilon(\tau=0,r)=\varepsilon_0, \quad \nabla_r\varepsilon(\tau,r=0)=0, \quad \varepsilon(\tau,r=a)=0. \end{aligned} \quad (1c)$$

The equations include transport terms both in the perpendicular (i.e. radial) and parallel directions with respect to the magnetic field lines. The perpendicular transport is assumed to be diffusive and governed by coefficients $C_D^{\kappa,\varepsilon}\frac{\kappa^2}{\varepsilon}$. The parallel transport along the magnetic field lines is here simply modelled by a linear loss operator controlled by the parameters $\sigma_\parallel^{\kappa,\varepsilon}$. On the right hand side the local drives of the turbulence are expanded up to order 2, with a linear growth of rate γ_s and a quadratic restoring term, in each equation. Finally, the term $-\varepsilon$ in the κ-equation identifies the field ε as a rate of damping for the turbulent energy κ.

The linear growth rates γ_s^κ and γ_s^ε are labelled s to underline that they are the source term for turbulence. In[14] they are equal and directly given by the linear interchange instability growth rate γ_I. Here we will include the parallel dynamic in effective growth rates as follows:

$$\gamma_\kappa = \gamma_s^\kappa - \sigma_\parallel^\kappa, \quad (2a)$$

$$\gamma_\varepsilon = \gamma_s^\varepsilon - \sigma_\parallel^\varepsilon. \quad (2b)$$

Either considering the hypothesis $\gamma_s^\kappa = \gamma_s^\varepsilon$ or not, we will assume $\gamma_\kappa \neq \gamma_\varepsilon$. Furthermore, the effective growth rates could become negative when the parallel stabilization $\sigma_\parallel$ exceeds the instability growth γ_s, but this case has not been considered yet, so we will suppose the growth rates positive in the following.

The quadratic term of the equation for κ insures that the field has no fixed point at infinity, preventing the divergence of the local dynamic behavior. It is weighted by the inverse of the parameter D_ω, which is found to be homogeneous to a viscosity. The subscript ω indicates that it can be linked to a typical width of the turbulent energy spectrum[1418]. Similarly, the nonlinear term in the equation for ε insures that the dissipation rate of κ eventually decreases when κ tends to 0. Its exact definition enforces that the fixed points κ_* and ε_* respect $\varepsilon_* = \kappa_*^{3/2}$, a typical scaling law of the Kolmogorov turbulence theory. It is weighted by the parameter V, homogeneous to a velocity. A closure using a scaling law for the SOL width gives $V \approx \rho_* c_s$ with ρ_* the Larmor radius normalised by a and c_s the sound velocity. Finally, two dimensionless coefficients C_D^κ and C_D^ε allow to tune the importance of the diffusion with regards to the local dynamic, possibly differently for κ and ε.

B. Normalised model

The normalisation of the $\kappa - \varepsilon$ system (Eq. 1) aims to isolate the critical parameters that determine the behaviour of the system, as it will be studied in detail in Section IV. A natural approach is to use the scaling given by the fixed point of the local drive of the system (right hand side of Eq. 1). We notably want to insure that for marginal growth rate $\gamma_\kappa \to 0$ and $\gamma_\varepsilon \to 0$ the fixed point of the reference turbulent viscosity $D = \kappa^2/\varepsilon$ vanishes.

We assume that the behaviour near marginality is governed by the dimensionless function g:

$$\gamma_\kappa = \gamma_0 \gamma_k \, g, \quad (3a)$$

$$\gamma_\varepsilon = \gamma_0 \gamma_e \, g^e, \quad (3b)$$

$$V = V_g V_0 \, g^v. \quad (3c)$$

Here γ_k, γ_e and V_g are dimensionless, γ_0 being the normalizing growth rate, and V_0 the normalisation of the parameter V. The non-zero fixed point κ_*, ε_* is then determined by:

$$\varepsilon_* = \gamma_e \gamma_0 \, g^{e-v} \, \frac{\kappa_*^{3/2}}{V_0 V_g}, \quad (4a)$$

$$\varepsilon_* = \kappa_*\left(\gamma_k \gamma_0 \, g - \frac{\kappa_*}{D_\omega}\right). \quad (4b)$$

We then recover the second order equation determining $\kappa_*^{1/2}$:

$$\left(\kappa_*^{1/2}\right)^2 + g^{e-v}\,\frac{D_\omega \gamma_e \gamma_0}{V_0 V_g}\,\kappa_*^{1/2} - g\,\gamma_k \gamma_0 D_\omega = 0. \quad (4c)$$

The positive root is therefore:

$$\kappa_*^{1/2} = g^{e-v}\,\frac{D_\omega \gamma_0 \gamma_e}{2V_0 V_g}\left[-1 + \left(1 + g^{1+2v-2e}\frac{4\gamma_k V_0^2 V_g^2}{\gamma_0 \gamma_e^2 D_\omega}\right)^{1/2}\right], \quad (5a)$$

$$\kappa_*^{1/2} \approx g^{1+v-e}\,\frac{\gamma_k}{\gamma_e} V_g \, V_0. \quad (5b)$$

The approximation used here is the limit $g^{1+2v-2e} 4\gamma_k V_0^2 V_g^2/(\gamma_e^2 \gamma_0 D_\omega) \ll 1$. We also obtain:

$$\varepsilon_* \approx \gamma_k \gamma_0 \, g^{3+2v-2e}\left(\frac{\gamma_k}{\gamma_e} V_g \, V_0\right)^2. \quad (5c)$$

One can then also determine $D_* = \kappa^2/\varepsilon$ for this fixed point:

$$D_* = \frac{\kappa_*^2}{\varepsilon_*} = \frac{V_0 V_g}{\gamma_0 \gamma_e}\, g^{v-e}\, \kappa_*^{1/2} \approx \frac{V_g^2 \gamma_k}{\gamma_e^2}\, g^{1+2v-2e}\frac{V_0^2}{\gamma_0}. \quad (6)$$

We remark that the condition used for the approximation can be written as $D_* \ll D_\omega$, or equivalently $D_0 \ll D_\omega$ with $D_0 = \frac{V_0^2}{\gamma_0} g^{1+2\nu-2e}$ the scale of D. We interpret it as the choice of a weak turbulence regime, which seems sufficient to match the results of the experiments[14]. We can quantify this condition with the ratio $K = D_0/D_\omega$, which we can be interpreted as a Kubo number[19][20] as it governs the transition from weak to nonlinear turbulence.

We may hence determine the condition on the exponent ν such that for $g \to 0$, one has $\varepsilon_* \to 0$, $\kappa_* \to 0$ and $D \to 0$. One readily finds that the condition $\kappa_* \to 0$ enforces $\varepsilon_* \to 0$, and that the condition $D \to 0$ enforces $\kappa_* \to 0$. The constraint $\nu > e - 1/2$, on the exponent of g in the expression of D given in (6), ensures that κ, ε and D exhibit the right limit as the growth rates approach zero. For the standard case $e \approx 1$, one finds therefore that $\nu > 1/2$ must be enforced, what we will achieve setting $\nu = 1$, as well as $e = 1$. We will even go a step further to simplify the discussion on the impact of the parameters and their identification, setting for the rest of this article $V_g = \gamma_k$. With those simplifications, we obtain the following scales from the local fixed points ((5b), (5c) and (6)) :

$$\kappa_0 = V_0^2, \quad \varepsilon_0 = \gamma_0 V_0^2, \quad D_0 = \frac{V_0^2}{\gamma_0} \tag{7}$$

We now set $Z = \kappa/\kappa_0$ and $Y = \varepsilon/\varepsilon_0$ and normalize the radial position by the system size, typically the plasma minor radius a, hence $\rho = r/a$, thus accounting for radial boundary conditions. Time is normalized by $1/\gamma_0$, $t = \gamma_0 \tau$ where γ_0 is the reference growth rate used in (3). We also normalise the growth rates : $\gamma_Z = \gamma_\kappa/\gamma_0$ and $\gamma_Y = \gamma_\varepsilon/\gamma_0$. With these definitions and the chosen scales (7), the system (1) now reads:

$$\partial_t Z = \frac{1}{\rho}\nabla_\rho\left(\rho D_{gBZ}\frac{Z^2}{Y}\nabla_\rho Z\right) + \gamma_Z Z - KZ^2 - Y, \tag{8a}$$

$$\partial_t Y = \frac{1}{\rho}\nabla_\rho\left(\rho D_{gBY}\frac{Z^2}{Y}\nabla_\rho Y\right) + \gamma_Y Y - \gamma_Z\frac{Y^2}{Z^{3/2}}, \tag{8b}$$

$$\begin{aligned} &Z(t=0,\rho) = Z_0, \quad \nabla_\rho Z(t,\rho=0) = 0, \quad Z(t,\rho=1) = 0, \\ &Y(t=0,\rho) = Y_0, \quad \nabla_\rho Y(t,\rho=0) = 0, \quad Y(t,\rho=1) = 0. \end{aligned} \tag{8c}$$

With $K = D_0/D_\omega$ introduced earlier, the only new symbols here are the diffusion weights $D_{gBZ} = C_D^\kappa/(\gamma_0 a^2)$ and $D_{gBY} = C_D^\varepsilon/(\gamma_0 a^2)$. Since one expects V_0 to scale like the normalised Larmor radius ρ_*, one then finds $D_{gBZ} \propto D_{gBY} \propto \rho_*^2$, which corresponds with the so-called gyro-Bohm (gB) scaling. System (8) will be the formulation used for the parameter identification because all parameters are decoupled, but an alternative notation will be interesting to study the behaviour of the system, setting $\gamma = \gamma_Z$, $u = \gamma_Y/\gamma_Z$, $D_{gB} = D_{gBZ}$ and $\Delta_Y = D_{gBY}/D_{gBZ}$. We get :

$$\partial_t Z = \frac{1}{\rho}\nabla_\rho\left(\rho D_{gB}\frac{Z^2}{Y}\nabla_\rho Z\right) + \gamma Z - KZ^2 - Y \tag{9a}$$

$$\partial_t Y = \frac{1}{\rho}\nabla_\rho\left(\rho \Delta_Y D_{gB}\frac{Z^2}{Y}\nabla_\rho Y\right) + u\gamma Y - \gamma\frac{Y^2}{Z^{3/2}} \tag{9b}$$

In the next Section III, we present the algorithm that will be used to identify those parameters given a target trajectory for Z and Y.

III. DATA ASSIMILATION ALGORITHM

The fitting procedure is only tested on twin experiments, fairly virtual cases where the target data are generated by the model itself, for some parameters that we try to find back. Although avoiding the problem of uncertainty and unavailability of the observations, it is already a sufficient test to give insights on the good behaviour of the algorithm and its ability to detect more or less complex sets of parameter values.

A. Definition of the problem

The target data are constituted of two sets of vectors Z_i^{obj} and Y_i^{obj} (the exponent obj stands for objective) containing approximations of Z and Y at different radial positions for a given set of times $(t_i)_{i\in[\![0,N_T]\!]}$ separated by the data time step δ_t^{obj}. This data time step is a multiple of the time step (generally 10 times bigger) used in the numerical solver to generate the data. Since the data are generated here, they are already complete, in the right format, and we do not have to take into account possible differences of confidence on the different observations. With real data, those problem will be mostly treated in the definition of the cost function, including operators to get the generated trajectory in the format of the data and adding a covariance matrix in the definition of the cost to account for observation errors. Here, we simply use an Euclidian scalar product of the difference between the logarithm of the generated trajectory and the data. The logarithm allows to consider a relative error which is more meaningful here because the variable can have important scale variations (see for instance in Section IV Figures 12 and 13). For a given set of parameters p, knowing the target set of vectors (Z_i^{obj}) and (Y_i^{obj}) corresponding to times $(t_i)_{i\in[\![0,N_T]\!]}$ the cost functional simply reads:

$$\begin{aligned} j(p) = \sum_{i\in[\![0,N_T]\!]} \frac{1}{2}\Big(&||\log((Z(p))(t_i)) - \log(Z_i^{obj})||_2^2 \\ &+ ||\log((Y(p))(t_i)) - \log(Y_i^{obj})||_2^2\Big) + \Pi(p), \end{aligned} \tag{10}$$

where Π is a regularising penalization term, depending directly on the parameters.

Concerning the parameters to identify, they are 5 in the model given in (8), namely D_{gBZ}, D_{gBY}, γ_Z, γ_Y and Z. We add to these parameters the initial states $Z(t_0)$ and $Y(t_0)$, simply written Z_0 and Y_0. In our case, the initial states are directly given by the first vectors in the data sets, Z_0^{obj} and Y_0^{obj}. However, even if the target initial conditions are given in the data, it is not obvious that the algorithm will easily converge towards them : at least in an intermediate state of the optimisation algorithm, it may be more advantageous to have the initial state

different from the one given by the data if it allows a reduction of the distance to the data as a whole. For real physical problems the data will most likely not be available at any point in space, and not directly for the turbulent variables κ and ε, so it is important to show that the initial condition can be identified.

All parameters are considered constant in time -as said above- and dependent on the radius, except D_{gBZ} and D_{gBY} which are simple scalars. Although this decision makes sense considering that the whole point of the model is to evaluate the reference viscosity, so correcting it with an ad-hoc parameter at every space point would clearly reduce its interest, it appears that the gradient with regards to this parameter heavily varies in scale with the discretisation points, making the problem poorly conditioned and the convergence very slow. Hence D_{gBZ} and D_{gBY} are simply considered as scalars.

Summing up, we have 7 different parameters (including the initial conditions), among which 5 vary with the radius, and all of them are used as unknowns of the inverse problem :

- D_{gBZ} and $D_{gBY} \in \mathbb{R}^+$ the coefficients of the (nonlinear) diffusion term, necessarily positive for a physically coherent diffusion effect.
- γ_Z and $\gamma_Y \in L^\infty([0,1] \to \mathbb{R}^+)$ the normalised effective growth rates. They may vary consequently in scale and could theoretically change of sign, but this case is not considered here, even if the PDE solver can marginally converge for negative values.
- $K \in L^\infty([0,1] \to \mathbb{R}^+)$ the Kubo parameter (also referred to the Strouhal number) weighting the nonlinear saturation term in the equation (8) for Z. The value of K is usually fairly negligible between 10^{-1} and 10^{-4}, but it is supposed to remain positive to act as a saturation term.
- Z_0 and $Y_0 \in L^\infty([0,1] \to \mathbb{R}^+)$ the initial state of the model. They are supposed positive due to the physical definition of Z and Y -and the mathematical operations performed on them make this condition necessary.

Despite the boundaries defined here, most of the parameters are allowed to take negative values at intermediate states of the minimisation algorithm, because it improves the convergence of the chosen minimizer. However their final retrieved values should indeed be positive.

B. Model solver and discretisation

A partially implicit time discretisation is used in the parameter fitting procedure to compute the trajectory of the normalised κ and ε to insure the stability for any D_{gB} with a relatively large time step, allowing less iterations in the algorithm. Actually, only the linear part of the diffusion term is implicitly computed:

$$Z_{n+1} - Z_n = \delta_t \left(\frac{D_{gBZ}}{\rho} \nabla_\rho \left(\rho D_{Z_n,Y_n} \nabla_\rho Z_{n+1} \right) + \gamma_Z Z_n - K Z_n^2 - Y_n \right), \tag{11a}$$

$$Y_{n+1} - Y_n = \delta_t \left(\frac{D_{gBY}}{\rho} \nabla_\rho \left(\rho D_{Z_n,Y_n} \nabla_\rho Y_{n+1} \right) + \gamma_Y Y_n - \gamma_Z \frac{Y_n^2}{Z_n^{3/2}} \right). \tag{11b}$$

The implicit part only adds the inversion of a sparse matrix (tridiagonal due to the simple spatial discretisation). When the model reaches a time t_i corresponding with recorded data vectors Z_i^{obj} and Y_i^{obj}, the squared difference between the logarithm of the current value of Z and Y and of the data Z_i^{obj} and Y_i^{obj} is added to the cost function. The model solver is derived by the automatic differentiation tool `Tapenade`[21], mostly as a proof of concept. For this simple solver the derivative could have been written directly but an automatic differentiation tool will be quite useful when extending to more complete models with substantially more complex solvers.

The scheme is still not stable for any set of parameters : the model can have very fast dynamics which lead to the change of sign of its variable in one step of the scheme, and if either Z or Y is negative somewhere the model diverges. This is actually a good test because solvers are rarely expected to give exploitable results in every regime, so the minimisation algorithm must be able to cope with some forbidden sets of parameters.

Radial grid spacing	Time step	Time elapsed between two recordings
2.5×10^{-3}	5×10^{-3}	5×10^{-2}

TABLE I. Numerical parameters for the discretisation in the parameter fitting procedure

Table I displays the values of the discretisation parameters used for all tests of the parameter fitting procedure. The data is generated with the same time step as the one used by the model solver in the parameter fitting algorithm ($\delta_t = 5 \times 10^{-3} s$), but the solution is recorded every 10 iterations (each $5 \times 10^{-2} s$). Using the same time steps ensures that we effectively consider twin experiments.

Finally, an important factor for the efficiency of the algorithm is the length of the time interval in the cost function. Since the model is often locally oscillatory, as we will see in details in the next section, the objective data and the currently computed trajectory may quickly run out of phase even if the parameters are relatively close from their target values. Hence it is interesting to reduce the length of the time interval to at most a few oscillations, at least as long as the cost is still high, with the added benefit of a largely reduced computational cost at each iteration.

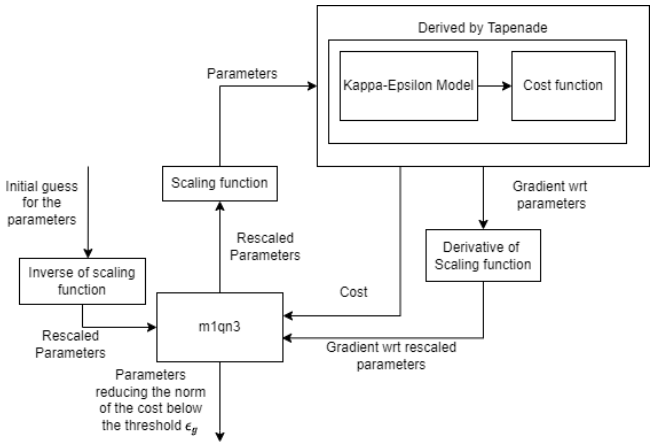

FIG. 1. Block scheme of the minimisation algorithm

C. Minimizer

The complete principle of the algorithm is illustrated in Figure 1. It is a loop between a nonlinear minimisation routine, the Fortran routine `m1qn3`[22], and the derived version of the whole procedure returning the cost, *i.e.*, the 1-D $\kappa-\varepsilon$ model solver slightly modified to compute the difference with the target data. Since the gradient is necessarily 0 at the minimum, the loop stops when the 2-norm of the gradient of the cost function has been enough reduced with regards to its initial value.

Most of the time the whole algorithm is actually launched two times, modifying between the two launches the penalisation function ($\Pi(p)$ in (10)) and the stopping criterion. The expected final reduction (after the last launch of the algorithm) of the gradient with respect to its initial value (before the first launch of the algorithm) is set to $\frac{||\nabla j||}{||\nabla j_0||} < \varepsilon_g = 10^{-7}$ (the stopping condition threshold).

The `m1qn3` routine implements a limited memory BFGS algorithm for unconstrained nonlinear optimisation (see[23]). It is a very efficient numerical minimiser, which converges quickly in terms of iterations, with a negligible cost in terms of computational time and memory space, compared to the evaluation of the cost and its gradient. One iteration of this minimiser can actually require multiple simulations, *i.e.*, evaluations of the cost function and of its gradient, but in general the number of simulations is rarely more than 30% larger than the number of iterations[23]. We use standard `m1qn3` parameters (see[22]). However the solver is only guaranteed to converge -without considering rounding issues- for an unconstrained variable, and we have multiple constraints on the parameters, either the necessary positivity of the initial states Z_0 and Y_0 or the impossibility to compute the cost and gradients for any arbitrary set of parameters. Thus two regularising strategies are used to avoid the divergence of the algorithm as well as to improve its general convergence.

a. Scaling Scaling functions are used to improve the conditioning of the problem and impose boundaries on some parameters. Each of the 7 parameters can have a different scaling function s, defined element-wisely for the non constant parameters. The scaling can be used to impose limits on a parameter: we just have to use a scaling function whose image is bounded, typically with a horizontal asymptote.

By default, all the parameters can have a linear scaling $x \mapsto kx$ of parameter k, which can be composed either to an exponential function, or what we choose to call the quasi-linear positive function[24]:

$$s : x \mapsto k\varepsilon \left(-\frac{1}{\pi} + \frac{1}{-\operatorname{atan}\left(\frac{x}{\varepsilon} + \frac{1}{\pi}\right) + \frac{\pi}{2}} \right) \tag{12}$$

This scaling function shown on Figure 2 is used to prevent a parameter from reaching 0 at some points, without adding too much non-linearity at the other points. The parameter ε is a good approximation of a threshold for the behaviour of the function.

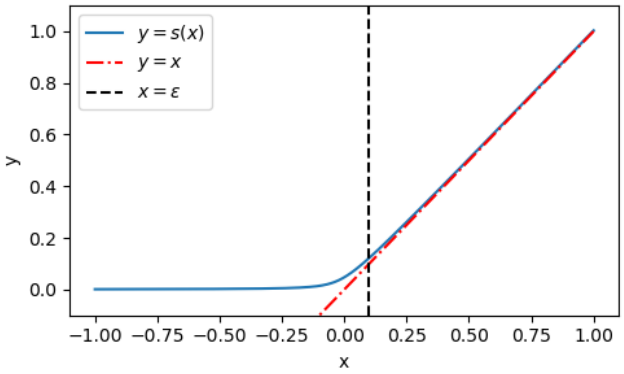

FIG. 2. Plot of the positive quasi-linear scaling function s for $\varepsilon = 10^{-1}$ and $k = 1$.

b. Penalisation Penalisation terms are added to the cost function to help keeping expected shapes and values for the parameters. They directly depend on the parameters and not on the result of the physical model. For each of them, a scalar weight balances its contribution to the cost. Two kinds of penalisation functions are used here on different test cases:

- Gradient penalisation: the quadratic norm of the gradient of the parameters with respect to the radius is added to the cost function.
- Negative value penalisation: since negative values in the parameters are either physically incoherent or may lead to the divergence of the numerical $\kappa-\varepsilon$ model solver, we will use in some cases a penalization function which simply adds to the cost one the square of the parameters at the points where they are negative.

IV. DYNAMICS OF THE $\kappa-\varepsilon$ SYSTEM

We study here the behaviour of the normalised $\kappa-\varepsilon$ system (Eq. 9) depending on the different parameters. In Sections IV B 1 and IV B 2, particular cases will be introduced

and used again as test cases for the parameter fitting procedure in Section 5.

A. Dynamics governed by the local drive

Without the diffusive transport term, the system (8) is reduced to a system of ordinary differential equations that writes:

$$\partial_{\gamma_Z t} Z = \gamma_Z\, Z - K Z^2 - Y, \tag{13a}$$

$$\partial_{\gamma_Z t} Y = \gamma_Y\, Y - \gamma_Z \frac{Y^2}{Z^{3/2}}. \tag{13b}$$

In the above model, the number of significant parameters is reduced by defining $\bar{Z} = \frac{Z}{\gamma_Z^2}$, $\bar{Y} = \frac{Y}{\gamma_Z^3}$, $\bar{K} = \gamma_Z K$, $u = \frac{\gamma_Y}{\gamma_Z}$ assuming that $\gamma_Z > 0$. Normalizing time by γ_Z finally leads to:

$$\partial_t \bar{Z} = \bar{Z} - \bar{K}\bar{Z}^2 - \bar{Y}, \tag{14a}$$

$$\partial_t \bar{Y} = u\, \bar{Y} - \frac{\bar{Y}^2}{Z^{3/2}}. \tag{14b}$$

One can consider the system Eq. 13 as a predator–prey system. For simplicity only in Section IV A, the bar is suppressed on the normalised variables and parameters.

1. Fixed points of the local evolution

The fixed points of the local system are solution of the coupled equations:

$$0 = Z_* - K Z_*^2 - Y_*, \tag{15a}$$

$$0 = u\, Y_* - \frac{Y_*^2}{Z_*^{3/2}}. \tag{15b}$$

Equation (15b) yields $Y_* = 0$ and $Y_* = u Z_*^{3/2}$. For $Y_* = 0$ one finds two possible solutions for Z_* in (15a), that are $Z_* = 0$ and $Z_* = 1/K$. For positive growth rates these points are not stable, so they do not really influence the system. For $Y_* \neq 0$, implying $u > 0$, Z_* is the positive solution of the second order equation $KZ + uZ^{1/2} - 1 = 0$ that we write as follows:

$$0 = Z_* + 2\frac{u}{2K} Z_*^{1/2} - \frac{1}{K}, \tag{16a}$$

$$Z_*^{1/2} = -\frac{u}{2K} + \sqrt{\frac{u^2}{4K^2} + \frac{1}{K}}. \tag{16b}$$

Analysing the dependency of the fixed point on the parameters K and u we are especially interested in the limit $K \to 0$, since it is the expected regime for the $\kappa - \varepsilon$ model, and we obtain

$$\lim_{K\to 0} Z_*^{1/2} = \frac{1}{u}. \tag{17}$$

Therefore $Z_* \approx 1/u^2$, and $Y_* = u Z_*^{3/2} \approx 1/u^2$. This holds so far that $u^2 \gg K$, as shown in the plot of u in the parameter plane u, K (see Figure 3).

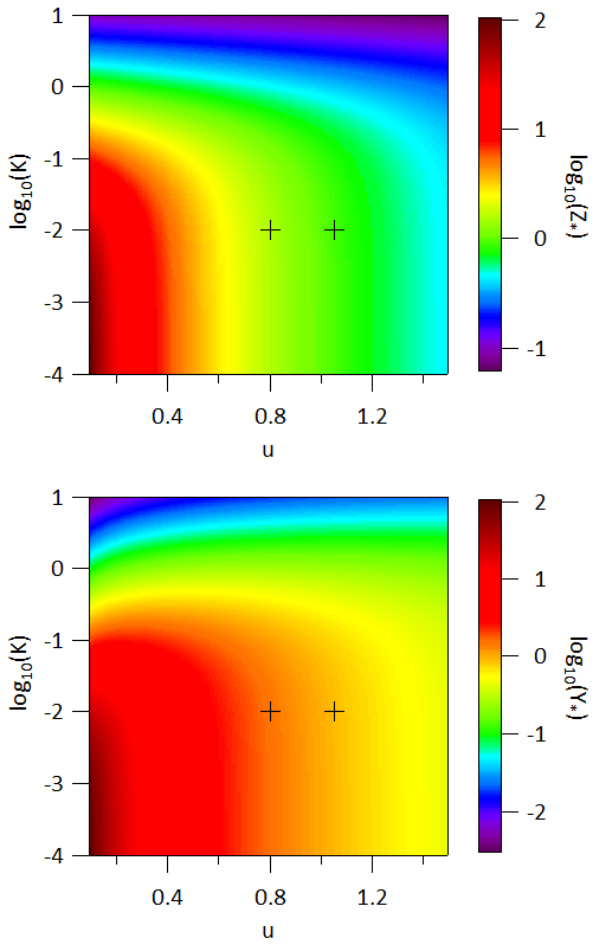

FIG. 3. Strictly positive fixed point in the control parameter plane u, K. Top: Z_*; bottom: Y_*.

Thus in the expected range for K, the normalised fixed point are mostly determined by the value of the growth rate ratio u. The actual fixed points are then obtained by multiplying Z by γ_Z^2 and Y by γ_Z^3, leading to the expected decrease with the value of γ_Z.

2. Tangential transform and asymptotic behaviour

We now examine the tangential transform obtained when linearizing equation (14) in the vicinity of any point (Z, Y) in the phase space.

$$\partial_t \widetilde{Z} = A\, \widetilde{Z} - \widetilde{Y}, \tag{18a}$$

$$\partial_t \widetilde{Y} = C\, \widetilde{Z} + B\widetilde{Y}, \tag{18b}$$

$$A = 1 - 2KZ\,;\; B = u - 2\frac{Y}{Z^{3/2}}\,;\; C = \frac{3}{2}\frac{Y^2}{Z^{5/2}}. \tag{18c}$$

The eigenvalues of this linear equation are solutions of the second order equation:

$$\lambda^2 - \lambda\,(A + B) + AB + C = 0. \tag{19a}$$

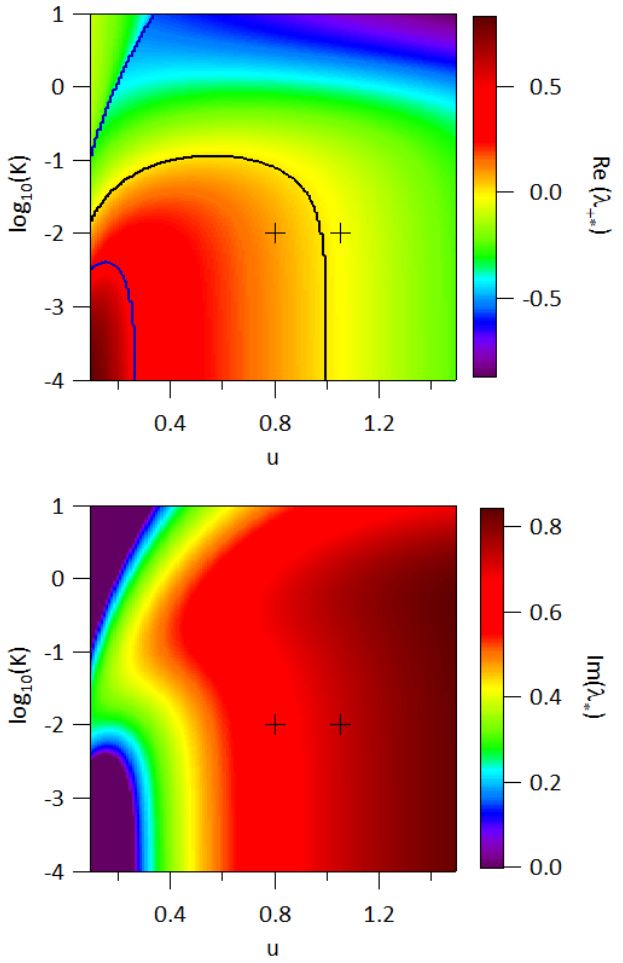

FIG. 4. Phase portrait of the fixed point eigenvalues. Real part (top) and imaginary part (bottom) of the largest eigenvalue.

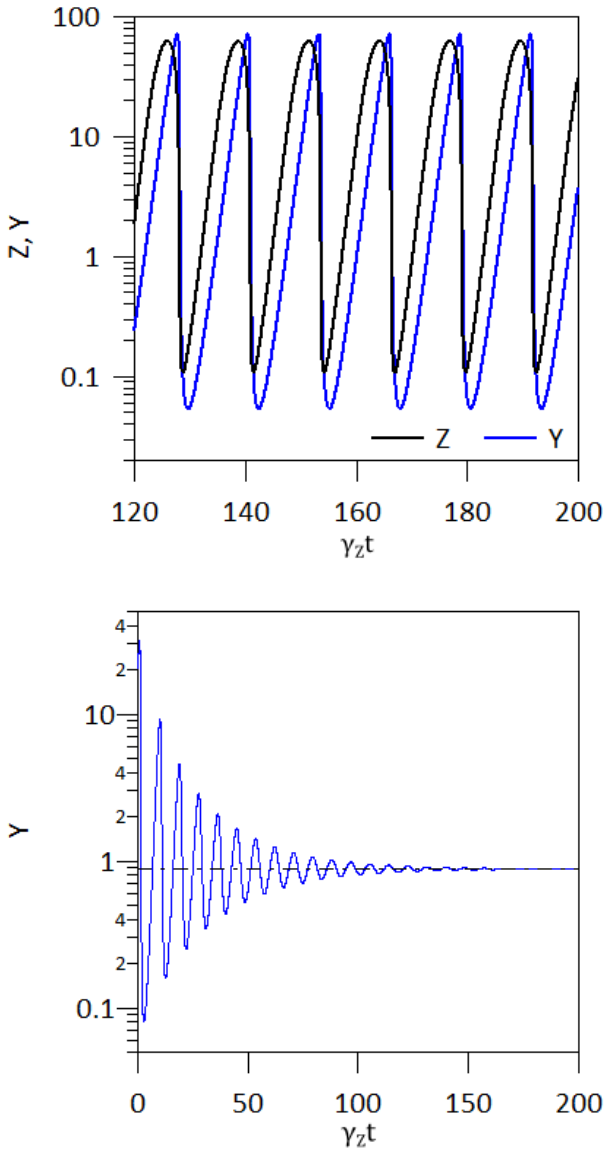

FIG. 5. Solution of the local system after the establishment of the stable limit cycle (logarithmic scale). Control parameters $K = 10^{-2}$, $u = 0.8$ (top) and $u = 1.2$ (bottom).

The canonical form of this equation is straightforward:

$$\left(\lambda - \frac{A+B}{2}\right)^2 = D = \left(\frac{A+B}{2}\right)^2 - AB - C, \tag{20a}$$

$$\lambda_\pm = \frac{A+B}{2} \pm D^{1/2}. \tag{20b}$$

From equation (20b) we then get the eigenvalues at the fixed points, which are indicative of the asymptotic behaviour of the system. Figure 4 displays the real part of the largest eigenvalue and the absolute imaginary part of both roots.

On the top part of Figure 4 the black curve is where the largest eigenvalue real part crosses 0, separating the zones of asymptotic stability and instability. In the expected regime with K small, the system becomes unstable for a value of u slightly smaller than 1. Concerning the imaginary part, it is mostly positive except for rather low values of u, leading to a mostly oscillatory asymptotic behaviour.

The time traces of the variables of the model on Figure 5 are coherent with the linear analysis. On the top for $K = 10^{-2}$ and $u = 0.8$, the system is asymptotically unstable, so it does not converge. The nonlinear terms prevent the system from reaching extreme values. After some time, the system reaches a limit cycle: first there is a phase of exponential growth with Y starting lower than Z until Z reaches a maximum value due to the saturation term $-KZ^2$; quickly after that, a brutal damping happens when Y becomes larger than Z and the coupling comes into play with the predator term $-Y$ in the equation for Z and then the sink term $-\frac{Y^2}{Z^{3/2}}$ of the equation for Y.

For $K = 10^{-2}$ and $u = 1.2$ (Figure 5 bottom), the model becomes asymptotically stable and quickly looks like the pseudo-asymptotic regime of a linear system, since in this range, the eigenvalues of the linearized system have non zero imaginary parts.

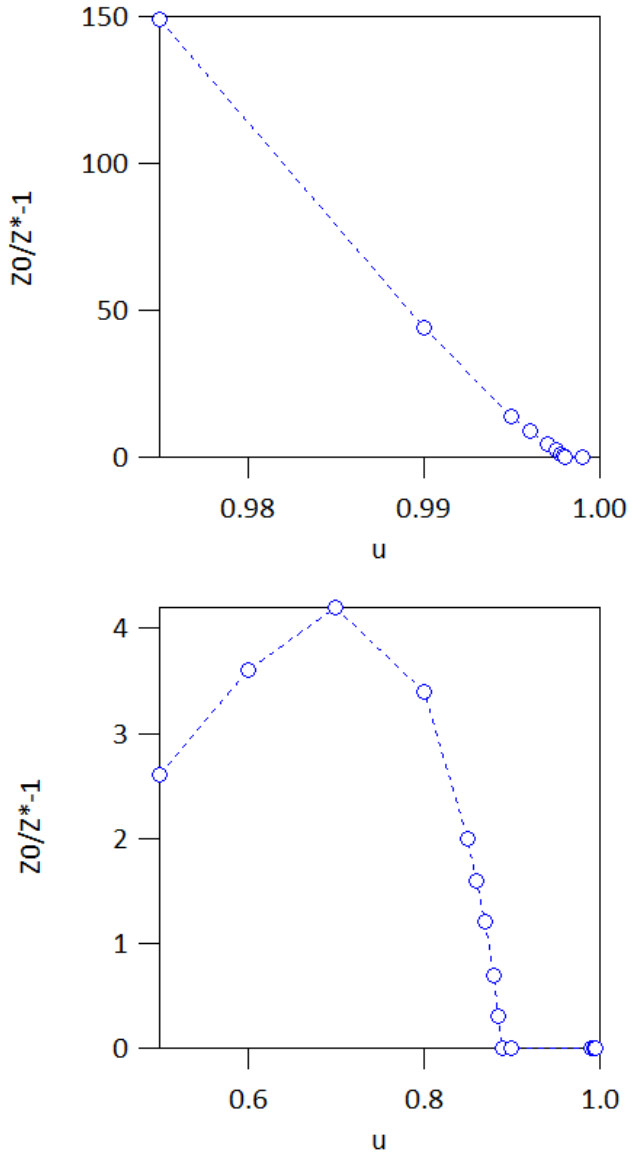

FIG. 6. Variation of the amplitude of the limit cycle when observing u for different values of K. (top) $K = 10^{-3}$ and (bottom) $K = 5.10^{-2}$.

3. *Amplitude and frequency of the limit cycles*

The linear analysis is not suited to give much information on the limit cycles since they are forced by the nonlinear terms of the system. It provides however a general idea of their dependency on the parameters that will be useful for the next sections.

Figure 6 shows the estimated amplitude of the limit cycle of Z with regards to the value of u for two different values of K. For the lowest value of $K = 10^{-3}$, it grows approximately linearly but very quickly to values that are two orders of magnitude higher than the fixed point. For $K = 5 \times 10^{-2}$, the growth is largely slower and reaches a maximum around $u = 0.7$. Indeed it can be expected to have the maximum limit cycle amplitude to scale as $\frac{1}{K}$ since it defines an actual limit for the value of Z (the derivative of Z is necessarily negative above this value):

$$Z - KZ^2 - Y \geq 0 \Rightarrow Z \leq \frac{1}{K}. \tag{21}$$

However this scaling law must be seen as a zero order approximation which does not take into consideration the coupling. Either way, big oscillations are at the same time physically questionable and quite challenging for the numerical solvers so we will keep $K = 5 \times 10^{-2}$ for most of the following experiments.

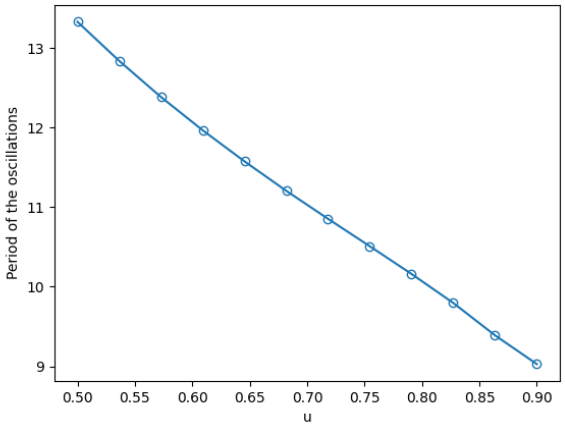

FIG. 7. Variation of the mean period of the oscillations of Z (in normalised time $\gamma_Z t$) for different values of u, with $K = 5 \times 10^{-2}$.

To conclude on the local system, let us just point out the influence of u on the period the limit cycles. Figure 7 shows for different values of u the mean period of oscillation of Z computed as the mean interval of time between two maximum values of Z between $\gamma_Z t = 100$ and $\gamma_Z t = 1000$ (to avoid the impact of the transient). The range of value of u gives almost exclusively a limit cycle behaviour. It is clear that the period decreases while u increases, contrary to the amplitude which starts to increase with u (Figure 6 bottom), and for this limited range, a $\frac{1}{u}$ dependency should give a good idea of the expected period value. Considering the parameter identification, if the frequency of the oscillations is dependent on the parameters, the result of the simulation for the current situation may have different local periods of oscillation from the ones of the target data. And for simulation lengths covering numerous periods the variables will eventually run out of phase, what will result in a considerable but not very informative contribution to the cost. In addition, local minima may appear when changing the frequency, if it maximises the number of regions where the simulation and the data are in phase during the simulation length. So the simplest solution is to use only a very short portion of the data, let us say about the length of a period of oscillation, to retrieve the parameters and avoid any potential issue of phase.

B. 1-D transport and spreading

Now we get back to the whole system including the diffusive transport term. The normalisation used for the local model is not applicable here if we want to be able to consider a variable γ_Z, so we get back to the reference formulation (8).

Before addressing 1-D simulations, we modify the expression of diffusive transport in cylindrical geometry to obtain a form more suitable for numerical implementation based on finite-difference derivatives:

$$\nabla_\rho\left(\rho D_{gBX} D_{Z,Y}\nabla_\rho X\right) = \rho D_{gBX} D_{Z,Y}\Big(\frac{1}{\rho} + \frac{\partial \mathrm{Log} D_{Z,Y}}{\partial Z}\nabla_\rho Z + \frac{\partial \mathrm{Log} D_{Z,Y}}{\partial Y}\nabla_\rho Y\Big)\nabla_\rho X + \rho D_{gBX} D_{Z,Y}\nabla_\rho^2 X. \quad (22)$$

Here X stands for either Z or Y depending on the chosen transport equation (and D_{gBZ}, D_{gBY} are constant). To ensure a good precision, in this section the simulations of the system (8) have been performed using a Runge-Kutta time stepping of order 4 and order 2 finite difference derivatives in ρ. Although we will use values inferior to 10^{-2} for D_{gBZ} and D_{gBY} (they should theoretically be of the order of ρ_*^2, with ρ^* the Larmor radius, hence much smaller than 1) the variable viscosity coefficient $D_{Z,Y} = \frac{Z^2}{Y}$ can take rather high value so the usual CFL condition (see, *e.g.*,[25]) for the diffusion has to be taken into account. Here, with δt and $\delta\rho$ as time and radial steps, it reads

$$\delta t < \frac{\delta\rho^2}{max(D_{gB} D_{Z,Y})}. \quad (23)$$

1. Spreading with 1-D variation of $u = \gamma_Y/\gamma_Z$

Spreading is one of the physical aspects where the κ-ε model can prove of particular interest compared to the standard quasi-linear approach to transport used in the fusion community[26]. This is also true for the dynamical aspect governed by the limit cycles. The set-up for the first series of simulations presented here is for constant $\gamma = \gamma_Z$ and K and a fixed profile of $u = \frac{\gamma_Z}{\gamma_Y}$ enforcing limit cycle for the local solutions except in the vicinity of $\rho = 0.55$ where the local analysis predicts convergence. The variation of u is governed by standard tanh-shaped step functions $S(\rho, \rho_b, \delta\rho_b)$ and window function $\Pi(\rho, \rho_{b1}, \delta\rho_{b1}, \rho_{b2}, \delta\rho_{b2})$ defined, respectively, as:

$$S(\rho, \rho_b, \delta\rho_b) = 0.5\left(1 + \tanh\left(\frac{\rho - \rho_b}{\delta\rho_b}\right)\right), \quad (24a)$$

$$\Pi(\rho, \rho_{b1}, \delta\rho_{b1}, \rho_{b2}, \delta\rho_{b2}) = 0.5\Big(S(\rho, \rho_{b1}, \delta\rho_{b1}) - S(\rho, \rho_{b2}, \delta\rho_{b2})\Big). \quad (24b)$$

For the present simulation we have thus set:

$$u = 0.75 + (u_t - 0.75)\Pi(\rho, \rho_{b1}, \delta\rho_b, \rho_{b2}, \delta\rho_b). \quad (25a)$$

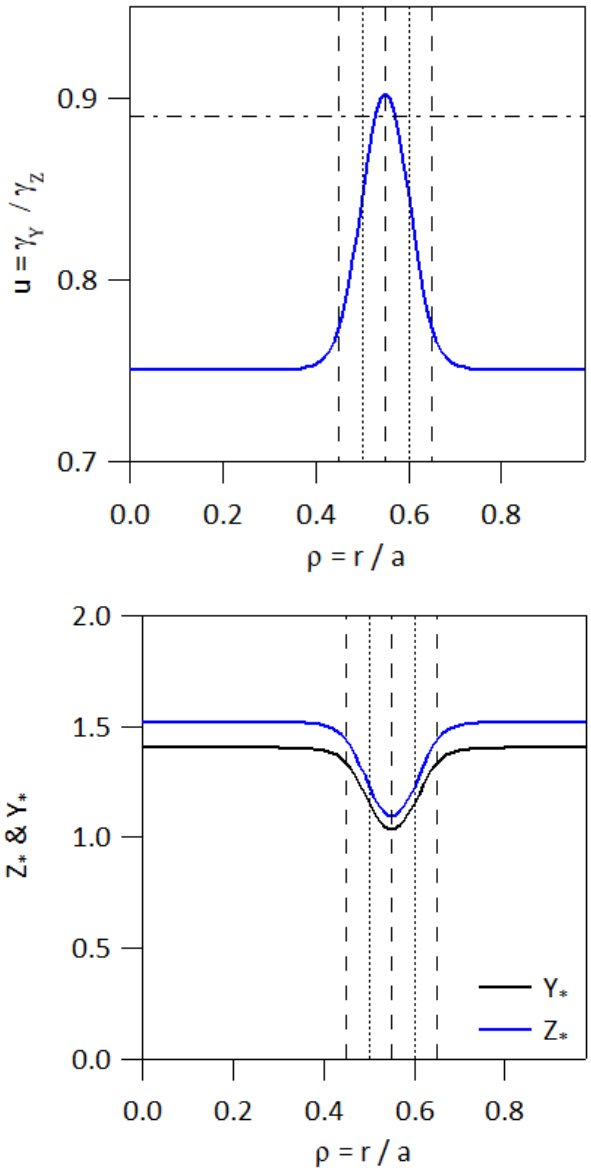

FIG. 8. Radial profile of u (top). Radial profiles of the fixed points Z_*, Y_* (bottom). The vertical dashed and dotted lines indicate the boundary of the tanh shaped transition region. The horizontal dash-dot line in the top figure indicates the critical value of u between the stable ($u \geq 0.89$) and the unstable ($u < 0.89$) fixed points.

The width of the two transition regions is chosen identical for both step functions, namely:

$$u_t = 1.15 \;;\; \rho_{b1} = 0.5 \;;\; \rho_{b2} = 0.6 \;;\; \delta\rho_b = 0.05. \quad (25b)$$

Since the distance $\rho_{b2} - \rho_{b1} = 2\delta\rho_b$ is small, the step function does reach its target value and u_t is adjusted to ensure that $u > 0.89$ in the window $0.528 \leq \rho \leq 0.572$, reaching $u \approx 0.9$ at $\rho = 0.55$ (see Figure 8 top). Given these parameters, the fixed point Z_*, Y_* of the local problem can be determined (see Figure 8 bottom). Since here $\gamma_Z = 1$ we can use directly the formula (16a). The y-scale is chosen to underline that the vari-

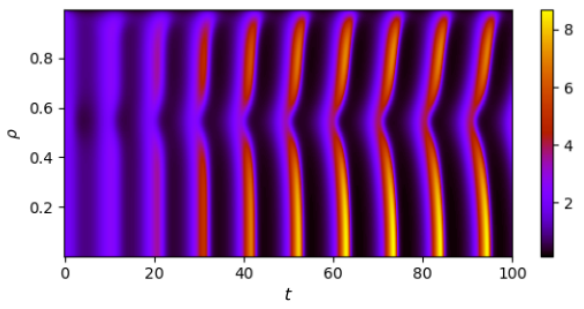

FIG. 9. Contour plot of Y versus time t and radial position ρ towards the beginning of the simulation for the profile of u shown in Figure 8, with $D_{gB} = 10^{-4}$, $K = 5 \times 10^{-2}$, and identical initial conditions at each point.

ation of Z_*, Y_* is small, in particular when compared to the amplitude of the limit cycles for $u = 0.75$.

A broad range of values of D_{gB} have been used for the simulations, from $D_{gB} = 10^{-10}$ to $D_{gB} = 10^{-1}$, always with $\Delta_Y = 1$, i.e. $D_{gBZ} = D_{gBY}$. Two effects become more and more evident as D_{gB} is increased (see Figure 9 with $D_{gB} = 10^{-4}$). First the absorbing condition at $\rho = 1$ obtained by enforcing $Z = 0$ and $Y = 0$ is now coupled with the other points in the radial profile governing a gradient in the values of Z and Y. The radial extent of this boundary layer is a measure of the spreading effect. Furthermore, after a transient period, the local oscillations have the same period at each radius, but with a certain delay constant in time. Even the region characterized by stable fixed points exhibits relaxation oscillations due to the radial coupling induced by the diffusion.

In the scan of the values of D_{gB} one can observe a gradual synchronization effect as D_{gB} is increased by a factor 10 from 10^{-9} to 10^{-5}. However the sharpest change and visible spreading of the relaxation oscillations into the stable region occurs between $D_{gB} = 10^{-5}$ and $D_{gB} = 10^{-4}$.

For values of D_{gB} lower than 10^{-5}, the diffusion is not intense enough to keep the value of Z and Y in phase at all the radii. For the lowest D_{gB} the diffusion even has an opposite effect, slightly spreading the stability zone in a region larger than the one suggested by the local analysis. The contour plot of Y for $D_{gB} = 4 \times 10^{-8}$ in Figure 10 is shown in grey with a reduced colour range from 0 to 3 to highlight the phase dynamics in the system and slightly zoomed in the radial direction $0.3 \leq \rho \leq 0.8$. The uniform grey shape between $\rho = 0.5$ and $\rho = 0.6$ is indeed where the model is locally converging, and it seems to reach out of the zone of local stability corresponding with the radius interval $[0.52, 0.58]$.

One can note that the phase difference grows up with time so that neighbouring points can exhibit opposite phases hence enhancing transiently the radial gradients. Hence for the latter times of the simulations, a resolution issue occurs even for quite precise radial discretisation, with a step of 2.5×10^{-3}. This would obviously get worse as D_{gB} decreases if the simulation was ran long enough. Either way there is no need to run the simulation very long to obtain good approximation of the parameters, so we will not need to reduce the radial discretisation in this case. In Section V we will show how the parameter fitting procedure handles the different behaviours linked to the different values of D_{gB}.

2. *Spreading with 1-D variation of γ with fixed $u = \gamma_Y/\gamma_Z$*

In this section we address the problem of spreading from a turbulent region with $\gamma_Y = \gamma_Z \approx \gamma_{ref} = 1$, into a region with reduced turbulence, typically $\gamma_Y = \gamma_Z \approx 0.1$. This is an interesting test for the effectiveness of spreading with fixed growth rate, akin to the assumption of fixed gradient sometimes used in tokamak plasma simulations[26].

We consider a reminiscent case of the so-called No Man's Land problem with three regions in the plasma, the core region with $\rho \leq 0.4$ that for simplicity we shall consider as turbulent with constant growth rate $\gamma = \gamma_{ref} = 1$, the gradient region with $0.4 \leq \rho \leq 0.75$ where the growth rate switches to $\gamma = 0.1$ and the edge and SOL region with $\rho \geq 0.75$ and $\gamma = 1$. Here the growth rate γ stands for $\gamma_Z = \gamma_Y$ since we consider $u = 1$. Changing u in this case has little effect on the important features of dynamical spreading and is therefore not addressed. A tanh dependence determines the shape of the variation of γ, with characteristic width parameter 0.03, yielding in practice to a transition region of width $\approx 2 \times 0.03$ as it can be see in Figure 11. The simulations are performed with Kubo number $K = 5 \times 10^{-2}$ and initial conditions determined by the local steady state Z_*, Y_*. The reference case is chosen with $D_{gB} = 10^{-3}$, hence a rather large value that is expected to enhance the spreading features as well as allowing for reduced simulation times since transport equilibrium is expected to occur after typically $1/D_{gB}$, hence a normalised time $t \approx 1000$ (the simulation time is set to half of this value). The evolution of the profile of Z normalised by the local fixed point value Z_* is shown on Figures 12 and 13 with $\log_{10}$ scale for $\Delta_Y = 2^{-6}$ (top) and $\Delta_Y = 1$ (bottom). In the core for both cases and edge regions Z hardly departs from the steady state value Z_*. As in the configuration of Section IV B 1, one can observe a localised decrease towards 0, the boundary value at the outer edge of the simulation region ($\rho \leftarrow 1$).

The main difference appears in the No Man's Land region. For $\Delta_Y = 1$, we have that Z mostly does not move from its initial value Z_*, and even decreases near the boundaries of the region so that there is no effective spreading from the surrounding more turbulent regions. On the contrary, for $\Delta_Y = 2^{-6}$, the value of Z is generally almost an order of magnitude larger than Z_* proving in this case an effective spreading. Moreover, Z exhibits large oscillations, localised towards the centre of the No Man's Land region, typically at $\rho \approx 0.7$, with a slowly damping amplitude. These oscillations are much slower than what could be found in the previous section, but this is still coherent with the local model. Indeed, a lower growth rate in the No Man's Land means that the time has to be dilated compared with the result of the local normalised model. Precisely, the time should be divided by the value of the growth rate, that here corresponds with a multiplication by 10, and the period goes approximately from 10 in the previous section to 50 here,

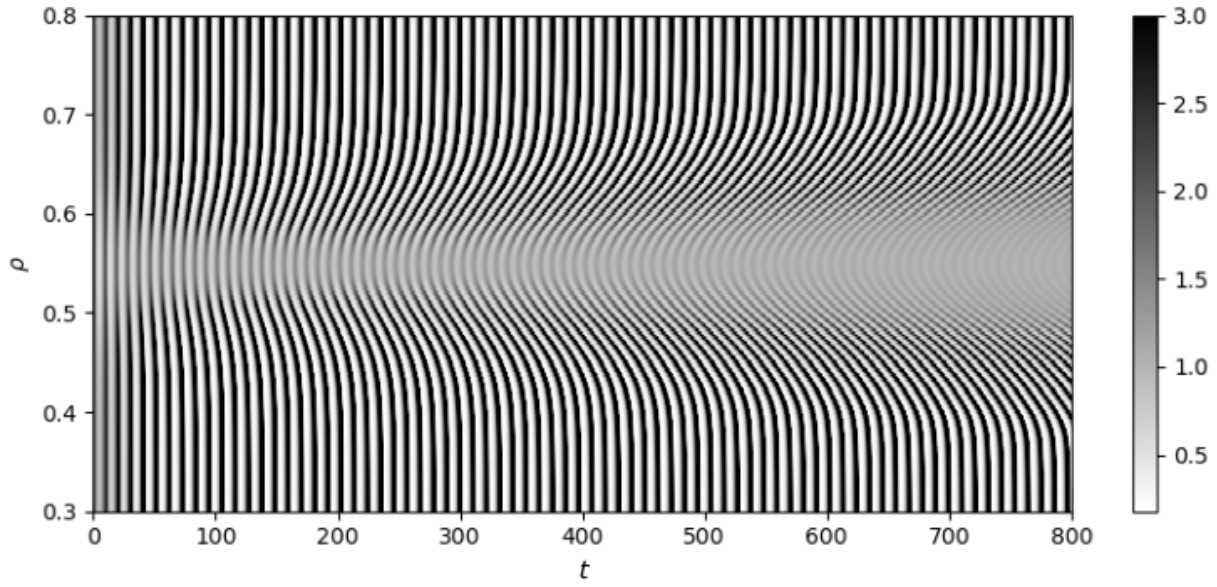

FIG. 10. Contour plot of Y versus time and radial position with a reduced colour range, with $D_{gB} = 4 \times 10^{-8}$.

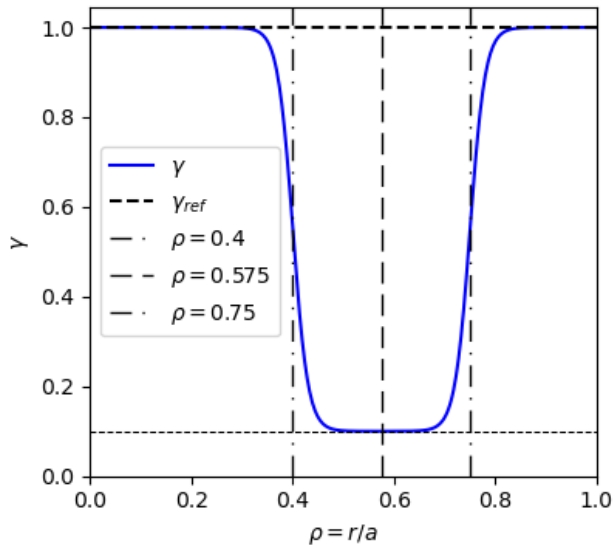

FIG. 11. Profile of the growth rate γ used in simulations of dynamical spreading. Dash-dot vertical line, boundary of the No Man's Land with a factor 10 reduction of the growth rate between $\rho = 0.4$ and $\rho = 0.75$, tanh transition with characteristic width 0.1, middle of this region at $\rho = 0.575$.

so the difference is not too far from expectation. Towards the two boundaries of the No Man's Land region steady state values appear to be reached on short simulation times, both for $\Delta_Y = 2^{-6}$ and $\Delta_Y = 1$.

The behaviour for $\log_{10}(Y/Y_*)$ in Figure 13 is generally qualitatively similar to the one of $\log_{10}(Z/Z_*)$ in both cases. Interestingly, for $\Delta_Y = 1$, the value of Y increases at the boundaries of the No Man's Land region, that leads to the decrease of the self generated viscosity proportional to $D_{Z,Y} = \frac{Z^2}{Y}$ (see Figure 14) and possibly explains the absence of spreading in this case.

Since $D_{Z,Y}$ is also proportional to the effective diffusion coefficient used in the fluid equations, its time average gives the intensity of the effective transverse transport caused by turbulences. Figure 14 compares the time averaged $D_{Z,Y}$ with its local fixed point value. For the standard case $\Delta_Y = 1$, the average transport $< D_{Z,Y} >_t$ is relatively close to the local fixed point, departing slightly on the borders of the No Man's Land region and around the edge boundary due to the homogeneous Dirichlet boundary condition. For $\Delta_Y = 2^{-6}$ however, while the average transport is lower than D_* everywhere else, it is consequently larger in the No Man's Land. In its middle, where $\gamma = 0.1$, one can even observe values that are larger than that of the core governed by $\gamma = 1$. One finds therefore that for $\Delta_Y = 2^{-6}$ the No Man's Land region generates a long transient with enhanced transport.

In order to investigate the properties of this dynamical spreading behaviour we first scan the value of Δ_Y given for $D_{gB} = 10^{-3}$. An average of $D_{Z,Y}$ from $t = 0$ to $t = 400$ is used to monitor the dynamical spreading (see Figure 15). One can notice that dynamical spreading increases as Δ_Y is halved, with $< D_{Z,Y} >_t$ reaching values 5 times greater at the center of the No Man's Land than outside, for the lowest value of the diffusion weights ratio. For $\Delta_Y > 2^{-2}$ the average $< D_{Z,Y} >_t$ remains approximately constant at D_*. It is to be noted that simulations with values of Δ_Y larger than 1 tend to be unstable with the chosen meshing and time stepping. The reduced transport of the predator Y therefore yields the expected effect of enhancing the transport features in the No Man's Land region. In the following, we will focus on cases with an ef-

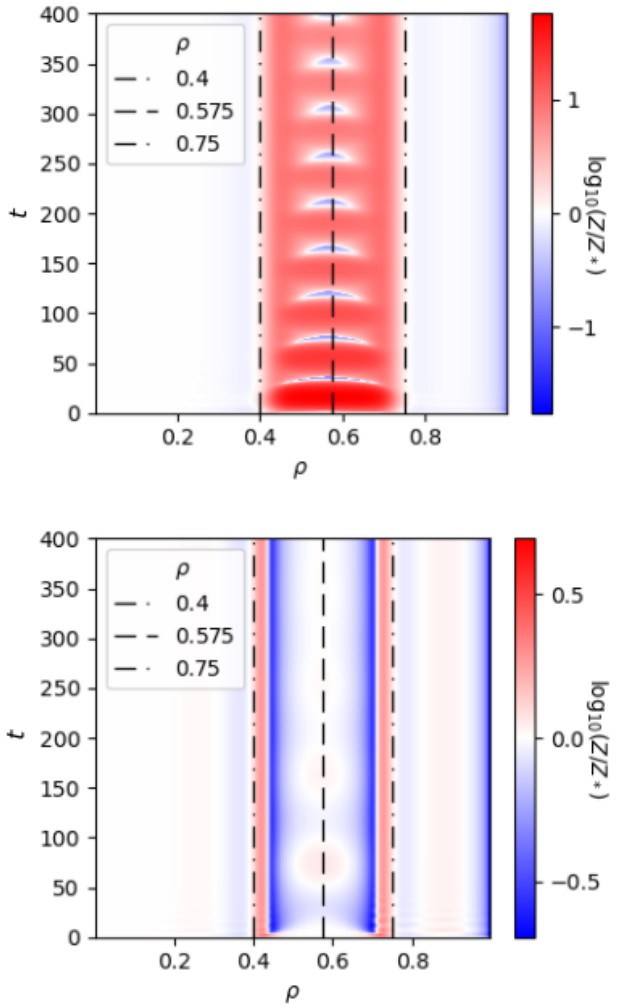

FIG. 12. Time evolutions of the profile of $\log_{10}(Z/Z_*)$ for $\Delta_Y = 2^{-6}$ (top) and $\Delta_Y = 1$ (bottom). Simulations are done with $D_{gB} = 10^{-3}$, $K = 5 \times 10^{-2}$ and the profile of γ shown in Figure 11.

FIG. 13. Time evolutions of the profile of $\log_{10}(Y/Y_*)$ for $\Delta_Y = 2^{-6}$ (top) and $\Delta_Y = 1$ (bottom). Simulations are done with $D_{gB} = 10^{-3}$, $K = 5 \times 10^{-2}$ and the profile of γ shown in Figure 11.

fective transport generated, that is by taking $\Delta_Y = 10^{-6}$ as a reference.

Investigating now the dependence of dynamical spreading on D_{gB}, two phases in the response can be identified, a first phase with an increase of the oscillation amplitude followed by the second phase with damping of the oscillations, as in Figure 16. There is an asymmetry of the amplitude with respect to Z_* with a larger increase for $Z > Z_*$ and a smaller decrease for $Z < Z_*$.

While the oscillation period exhibits a rather small dependence on D_{gB}, the increase and decay of the oscillation amplitude depends on D_{gB}. At small values of D_{gB}, typically 5×10^{-5}, the amplitude evolution is slow. For larger values of D_{gB} the amplitude variation in time is faster and the oscillation period becomes shorter. The onset of the oscillations is then rapid and only the decay phase can be identified, even though the maximum oscillation amplitude is still considerably larger for the highest than for the lowest D_{gB} values. Furthermore, at low values of D_{gB} the trajectories appeared to be cycling towards the local fixed point Z_*, for larger values of D_{gB} the convergence point clearly detaches from that fixed point. The behaviour of Y is qualitatively similar to that of Z and so is the one of $D_{Z,Y}$, although its detachment from the fixed point is less noticeable. Except for the highest values of D_{gB}, we have that $D_{Z,Y}$ converges to a value quite close to its local minimum so the increased transport is mostly a transient behaviour.

Finally we investigate the scan of D_{gB} using a different criterion to evaluate the dynamical spreading achieved with $\Delta_Y = 2^{-6}$, namely the maximum and minimum values of $D_{Z,Y}$, achieved during the first cycle at $\rho = 0.575$. Dynamical spreading behaves as a resonance when drawing the maximum of $D_{Z,Y}$, this value being reached at the beginning of the first relaxation cycle, versus D_{gB} (see Figure 17 top, in semi-log scale). The same data is presented in Figure 17 bottom with a log-log scale, top curve head-up triangles together with the minimum of $D_{Z,Y}$ during dynamical spreading, and the aver-

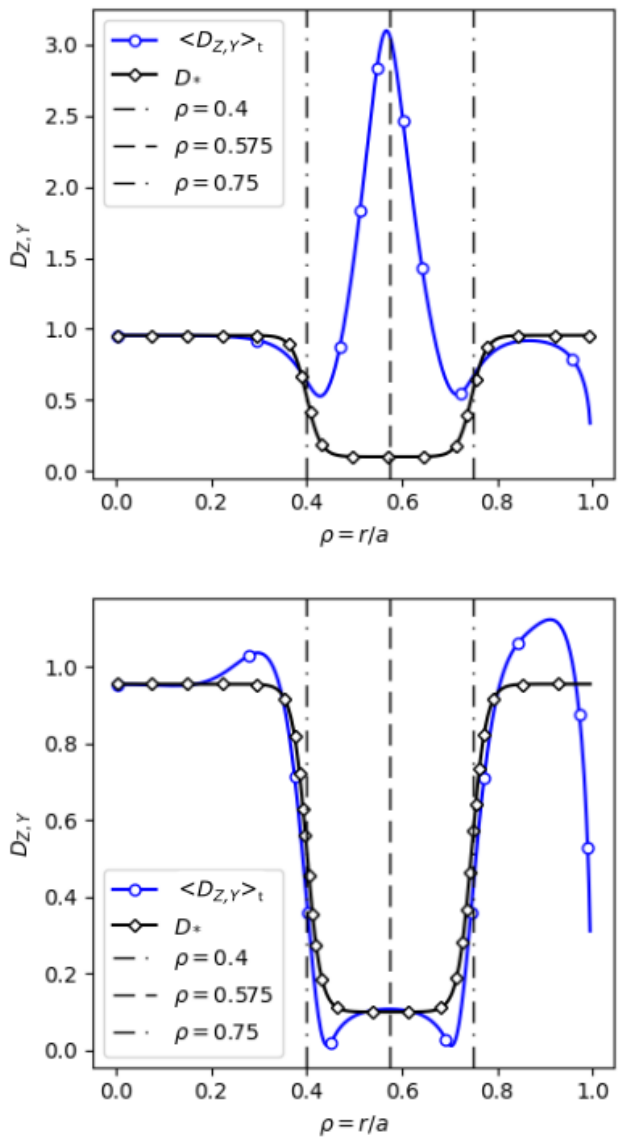

FIG. 14. Profiles of the time average value $< D_{Z,Y} >_t$ and that of the local fixed point D_*, for $\Delta_Y = 2^{-6}$ (top) and $\Delta_Y = 1$ (bottom). Simulations are done with $D_{gB} = 10^{-3}$, $K = 5 \times 10^{-2}$ and the profile of γ shown in Figure 11.

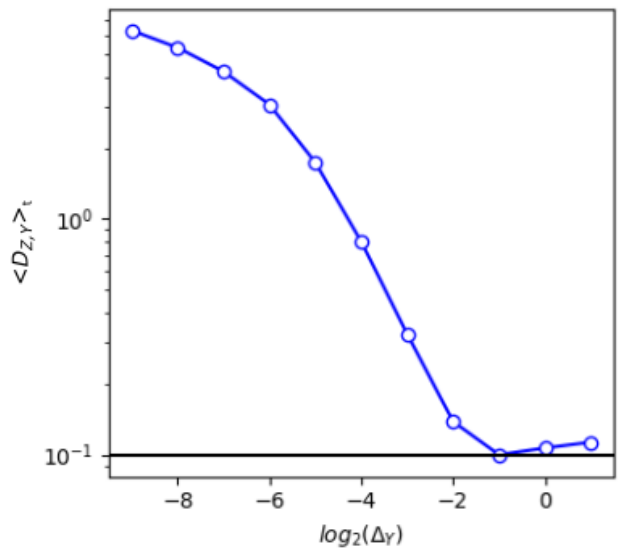

FIG. 15. Transition with Δ_Y of the transient enhancement of transport monitored by the time average of $D_{Z,Y}$ at $\rho = 0.575$ from $t = 0$ to $t = 400$, blue curve open circles. Simulation with $D_{gB} = 10^{-3}$, $K = 0.05$ and the profile of γ shown on Figure 11. Black line: D_* at the center of the No Man's Land ($\rho = 0.575$).

age in time from $t = 0$ to $t = 400$. The black dashed line represents the value of D_* at the same radial location $\rho = 0.575$. One finds that there is a threshold at low values for $D_{Z,Y}$ to detach from D_*. For these small values of the transport, the local dynamics govern the evolution of Z and Y and transport has little effect. Conversely, when D_{gB} is large, the damping rate of the relaxation events, that is observed to increase with D_{gB}, is fast enough to modify the initial cycles during which $\min(D_{Z,Y})$ and $\max(D_{Z,Y})$ are reached. The minimum of $D_{Z,Y}$ being reached at later times than the maximum is the first to exhibit a rapid switch-off. A similar sharp drop of the maximum would require a still larger value of D_{gB}.

We have analysed several cases where spreading takes place with the κ-ε model. It leads to a decoupling of the observed values of κ and ε from that determined by the local dynamics. The effect is governed by the self-consistent transport proportional to κ^2/ε. A consequence is that transport also deviates from the expected one. One finds therefore that even in this model with fixed growth rates, and thus akin to fixed gradient driven turbulence[26], spreading occurs. In the cases of interest for the analysis of plasma-wall interaction as well as for realistic core simulations one must consider the prescribed fluxes. Varying transport at fixed gradient can only hold with varying fluxes that adapt to the achieved transport magnitude, which is not relevant for steady state operation. Stepping to flux driven simulations, one can then expect that spreading will be more effective since the modified transport governed by the κ-ε will also modify the growth rate via the evolution of the gradients, hence generating a higher dimension phase space where nonlinear effects can drive novel transport phenomena and in particular enhanced spreading properties.

For now we will see how the parameter identification procedure behaves on the analysed cases, and how it is influenced by the weight of the nonlinear diffusion term.

V. PARAMETER IDENTIFICATION

This section aims to show the potential of the variational data assimilation procedure introduced in Section III to retrieve the parameters used in the two spreading regimes previously identified, starting from relatively standard initial guesses.

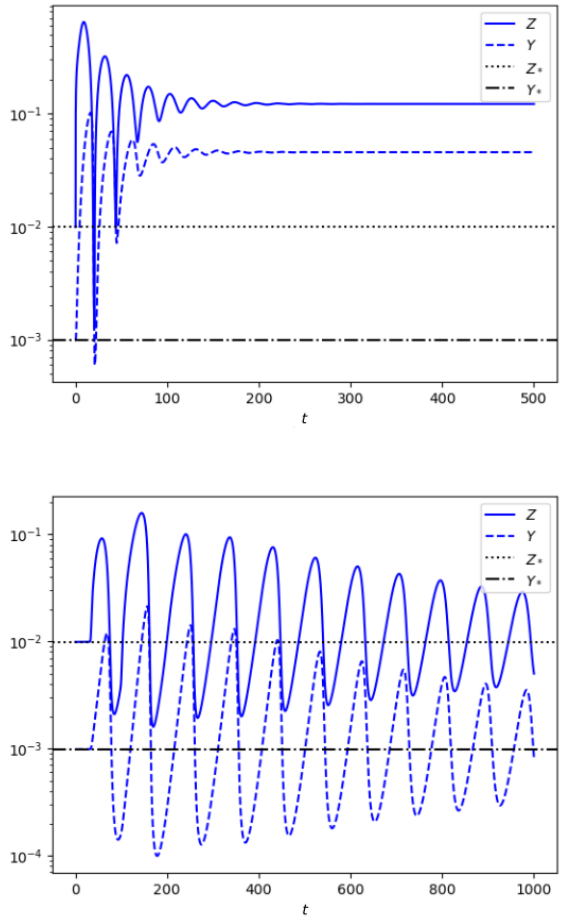

FIG. 16. Time evolutions of Z and Y at the center of the No Man's Land, with $D_{gBZ} = 7 \times 10^{-3}$ (top) and $D_{gBZ} = 5 \times 10^{-5}$ (bottom).

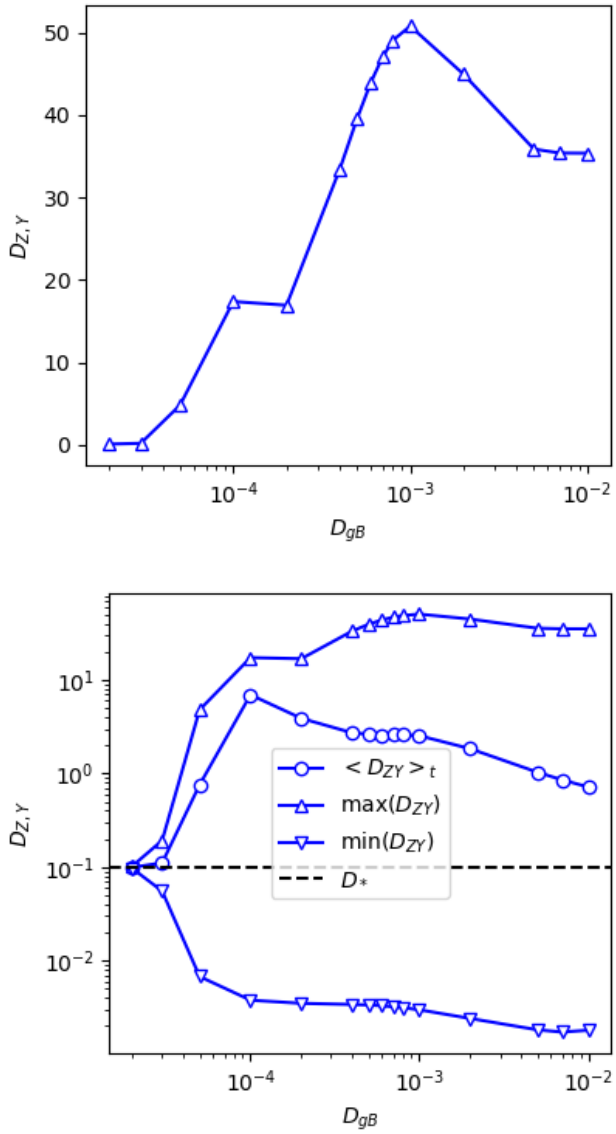

FIG. 17. Value of $D_{Z,Y}$ during dynamical spreading at $\rho = 0.575$. Maximum value of $D_{Z,Y}$, in semi-log scale (top). Maximum, minimum and average values of $D_{Z,Y}$, from $t = 0$ to $t = 400$, in log-log scale (bottom). The dashed black line indicates the value of D_*. Simulations are done with $\Delta_Y = 2^{-6}$, $K = 5 \times 10^{-2}$ and the profile of γ shown in Figure 11.

A. Test case A : spreading with variation of $u = \frac{\gamma_Y}{\gamma_Z}$

In a first set of tests, the aim is to recover the profile of $u = \frac{\gamma_Y}{\gamma_Z}$ (see Figure 18) used in Section IV B 1, assuming the other parameters constant in space. The minimisation is tested on 4 sub-cases where only the target value of D_{gB} (designated as D_{gB}^{obj}) is changed from 10^{-2} to 10^{-8} with $\Delta_Y = 1$ (i.e. each time $D_{gBZ}^{obj} = D_{gBY}^{obj}$) usually resulting in slightly different behaviours of the minimisation algorithm.

The parameters used in Section IV B 1 are mostly flat, so using the gradient penalization seems a little bit too favourable as it naturally pushes the parameters to constant values. Hence instead of trying to recover the flat initial condition for Z and Y used in Section IV B 1, we let the model run (for the specified target parameters) and use the shapes of Z and Y, obtained at a certain normalised time t_0, as the initial conditions Z_0 and Y_0 to recover. Hence, since the system is autonomous, the data generated for the assimilation procedure will be equal to those for the simulation initialized with flat initial conditions in the time interval $[t_0, t_0 + T]$, T being the length in time of the generated data. In the following we will use this time interval to designate what the data corresponds with, hence defining the target initial condition and its time length. For this first case we typically use the time interval $[40, 50]$, so that the initial condition to retrieve will be $Z(40)$ and $Y(40)$ for Y and Z started at $Z(0) = 2$ and $Y(0) = 2$, and the time length of the data is equal to 10. Figure 19 shows the shapes of the initial states that will be retrieved by the minimisation procedure, and the first guesses from where the algorithm starts.

The two last parameters are kept constant at their original value, that are $\gamma_Z = 1$ and $K = 0.05$. The initial guesses for

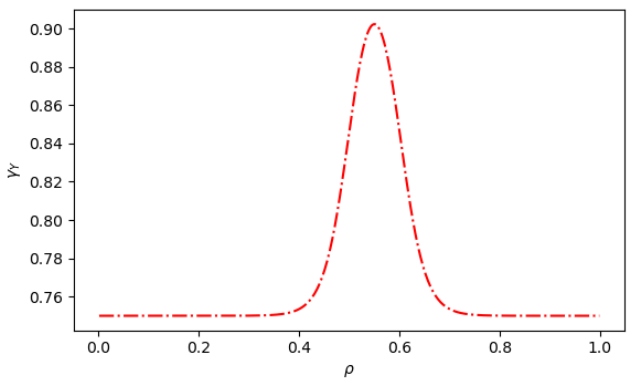

FIG. 18. Radial profile of the target γ_Y for the parameter fitting test case A.

	D_{gBZ}	D_{gBY}	γ_Z	γ_Y	K
Initial guess	10^{-5}	10^{-5}	1.2	0.7	2×10^{-2}
Target value	10^{-8} to 10^{-2}	D_{gBZ}	1.0	Gaussian shape	5×10^{-2}

TABLE II. Initial guess and target values of the parameters for the different configurations (except Z_0 and Y_0, for which we refer to Figure 19).

parameters other than Z_0 and Y_0 are also constant, with the same values, for each value of D_{gB}^{obj}. They are $D_{gBZ} = D_{gBY} = 10^{-5}$, $\gamma_Z = 1.2$, $\gamma_Y = 0.7$, $K = 2\times10^{-2}$. These values are chosen so that the first guess presents limit cycles as the target solution, even though this is probably not necessary for the convergence of the algorithm. These values are summarized in Table II.

Concerning the parameters internal to the minimisation algorithm, some scaling is performed either to introduce a limit on the value of a parameter, to adapt a gradient value largely different from the others or to introduce a rough idea of the proximity of the initial guesses to the target parameters. We notably used an exponential scaling for D_{gBZ} and D_{gBY} because the gradient with regard to these parameters is orders of magnitude larger than that with regard to the other parameters, and seemingly growing for lower values of D_{gBZ} or D_{gBY}. Furthermore, we only force, with a Positive Quasi Linear (PQL) scaling function, the positivity of the initial state Z_0 and Y_0 functions because these are the only parameters that must always remain positive. The minimisation algorithm generally behaves better without constraints on the parameters. The scaling are summarized in Table III.

	D_{gBZ}	D_{gBY}	γ_Z	γ_Y	K	Z_0	Y_0
type	exp.	exp.	lin.	lin.	lin.	PQL	PQL
linear coefficient	5.0	5.0	1.0	1.0	1.0	5.0	5.0
ε coefficient	N/A	N/A	N/A	N/A	N/A	10.0	10.0

TABLE III. Scaling functions on the different parameters for the test case A with the time interval [0,10]. Here, exp., lin. and PQL stand, respectively, for exponential, linear and positive quasi-linear.

D_{gB}^{obj}	w_p	Iterations	Simulations	Final cost
10^{-8}	0	403	412	2.39×10^{-12}
10^{-8}	10^{-2}	507	518	4.01×10^{-6}
10^{-6}	0	541	549	7.93×10^{-12}
10^{-6}	10^{-2}	545	560	2.44×10^{-5}
10^{-4}	0	441	542	2.56×10^{-6}
10^{-4}	10^{-2}	710	741	3.24×10^{-6}
10^{-2}	0	976	1060	1.92×10^{-8}
10^{-2}	10^{-2}	1070	1168	3.46×10^{-8}

TABLE IV. Number of iterations and final cost for the test case A with and without gradient penalization. The gray colour indicates that the minimiser did not converge because the line search failed.

Moreover, although the algorithm works relatively well without penalisation, we introduce a gradient penalisation to prevent the apparition of oscillations on the parameters, especially on the boundaries of the domain. Specifically, we run the minimisation algorithm twice in a row. A first time, a high weight for the penalisation $w_p = 10^{-2}$ and a relaxed stopping condition $\frac{\|\nabla j\|}{\|\nabla j_0\|} < \varepsilon_p^i = 10^{-5}$ are imposed to prevent the apparition of big oscillations when the parameters are far from their targets. Then, a second time, a lower weight $w_p^+ = 10^{-4}$ and the original stopping condition $\frac{\|\nabla j\|}{\|\nabla j_0\|} < \varepsilon_p = 10^{-7}$ are adopted mostly to accelerate the convergence in the cases where the shapes are harder to recover. The second step can be useful when dealing with real data, for which a very high precision cannot be obtained, so a quicker convergence toward a slightly flattened solution is desirable.

Table IV shows the number of iterations and the final cost for the different studied configurations, and Figure 20 shows the evolution of the cost function and norm of its gradient, versus the number of simulations for some chosen cases. As a reminder, each simulation consists of one computation of the cost function and of its gradient and there can be several of them in one iteration (although as seen in Table IV this is quite rare for a convergent case). First, let us remark that the cost for the case with penalisation has a very specific shape: we can see two times in a row a phase of quick decrease followed by a quasi stagnation of the cost. The stagnation is simply due to the fact that the penalisation function is not zero when the parameters are equal to the target parameters, so there is a threshold under which the cost cannot go. Then the shape is doubled because when the gradient penalization is used the minimisation algorithm is launched 2 times, with different penalization weights and stopping thresholds. The launch of the second minimisation procedure can also be spotted on the gradient plot where the gradient norm suddenly increases.

In each case the gradient norm is reduced, but the procedure is largely faster for the lowest D_{gB}^{obj} and, at equal D_{gB}^{obj}, slightly faster without gradient penalization. Furthermore, the cost function is logically more reduced without penalisation, but while the gap is significative for lower target D_{gB} it is largely reduced for higher ones. Furthermore, the minimiser does not converge for $D_{gB}^{obj} = 10^{-4}$ without the penalisation, because a satisfactory step cannot be found for the descent direction at

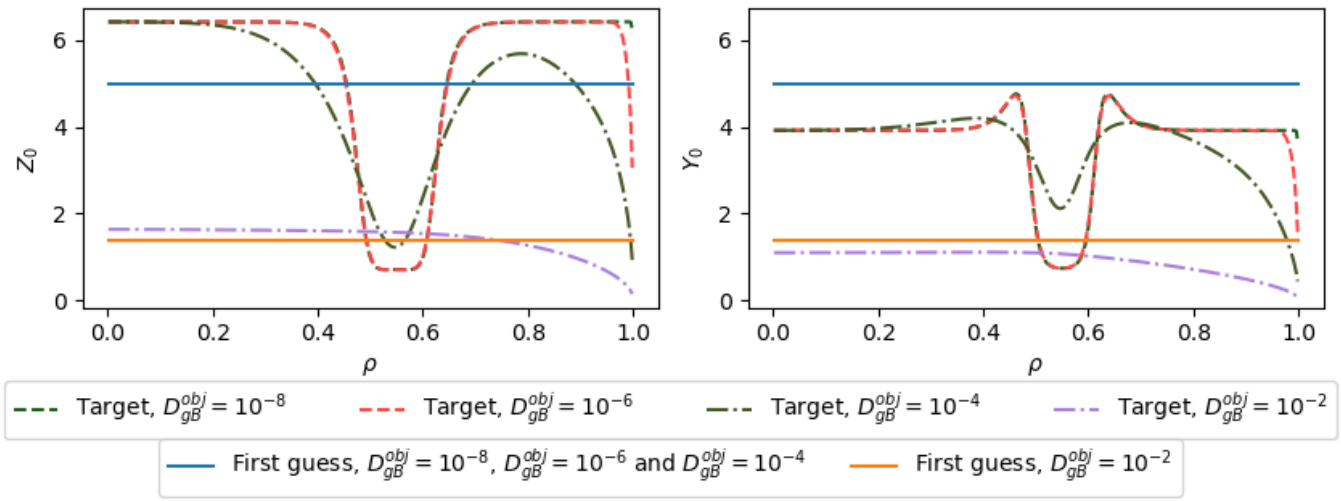

FIG. 19. Values of the target and first guesses for the initial states Z_0 and Y_0 for the different D_{gB}^{obj}. The target are obtained by running the simulation until $\gamma_0 t = 40$, starting with $Z(0) = Y(0) = 2.0$ for the given D_{gB}^{obj}.

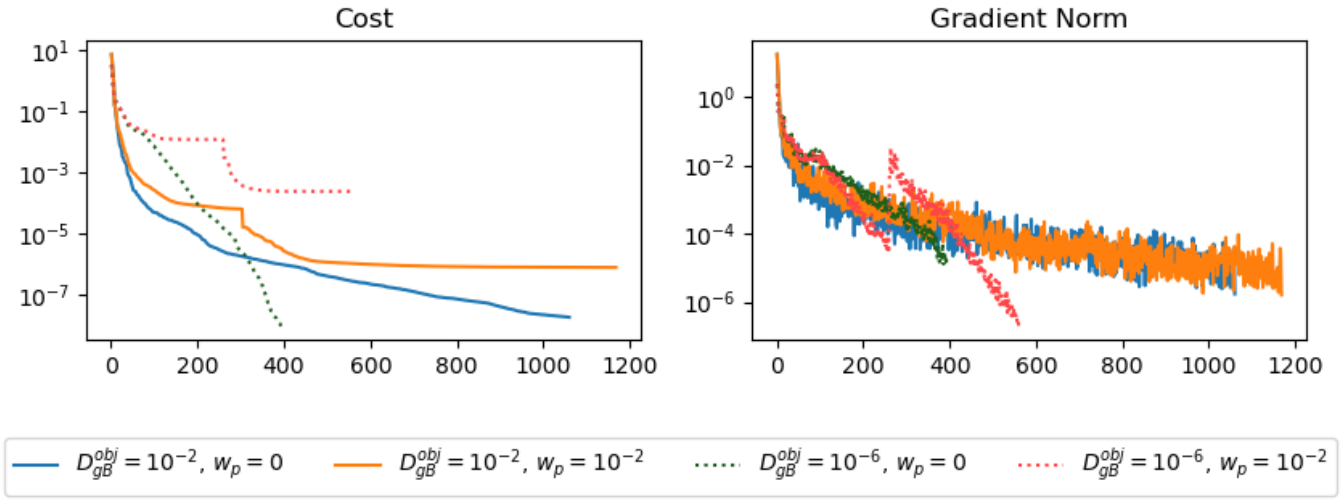

FIG. 20. Evolution of the cost and the norm of its gradient versus the number of iterations (computations of the cost function and of its gradient), for 2 different values of the target D_{gB} of the test case A, with and without gradient penalisation (w_p is the weight of the gradient penalization in the first run of the minimisation algorithm).

the last iteration. This happens relatively late in the simulation since the gradient norm is still considerably reduced. Let us see now how these observations are associated with the error on the parameters.

Table V shows the final relative error on the parameters and Figure 21 shows the relative error on the parameters, as functions of the number of simulations, for a chosen subset of configurations. When D_{gB}^{obj} is small, the algorithm quickly identifies γ, u, K, Z_0 and Y_0, and then it takes more iterations to identify D_{gBZ} and D_{gBY}. In the case $D_{gB}^{obj} = 10^{-8}$ with penalisation, the final value for D_{gBZ} is even still almost the quadruple of its target. The error could probably be corrected with a more precise stopping condition, but either way it shows that the identification of D_{gBZ} and D_{gBY} is quite difficult when their target value is low, simply because they have a low impact on the behaviour of the model, mostly driven by the local dynamic. Moreover, for lower target D_{gB} the minimisation seems clearly faster and notably more precise without the penalisation.

On the other hand, for higher target D_{gB} the identification of D_{gBZ} and D_{gBY} is relatively fast, but then the algorithm identifies very slowly -and with a low final precision- the parameters varying with space. A probable explanation is that the intense diffusion has a blurring effect on Z and Y so that the shapes of the parameters have less influence on the shapes of the variables, which makes them harder to identify.

Figure 22 completes the analysis by showing the profile of

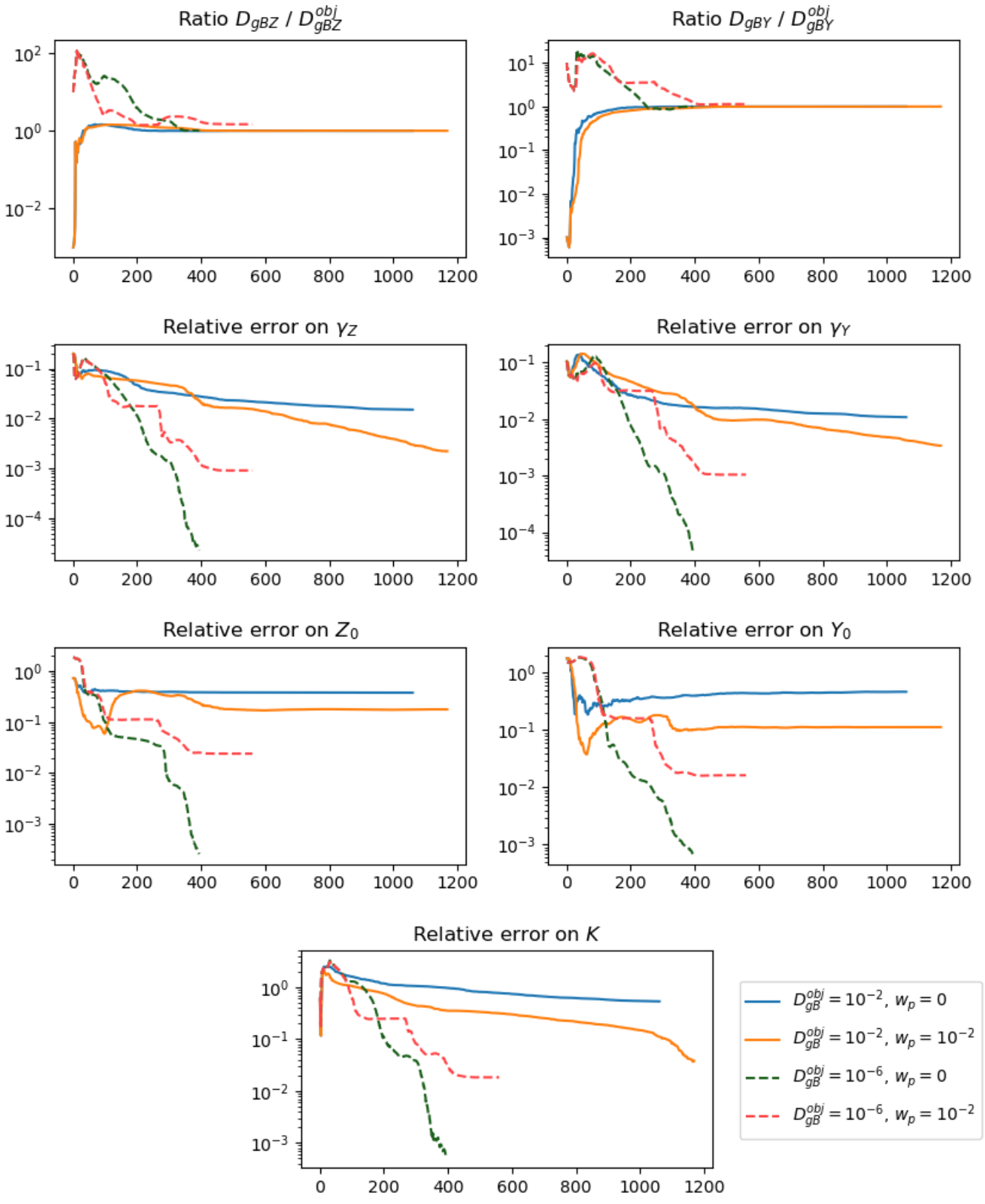

FIG. 21. Evolution of the relative error on the parameters versus the number of iterations (computations of the cost function and of its gradient), for 2 different values of the target D_{gB} of the test case A, with and without gradient penalisation (w_p is the weight of the gradient penalization in the first run of the minimisation algorithm).

D_{gB}^{obj}	w_p	D_{gBZ}	D_{gBY}	γ_Z	γ_Y	K	Z_0	Y_0
10^{-8}	0	2.42×10^{-3}	1.24×10^{-3}	1.05×10^{-6}	2.26×10^{-7}	9.27×10^{-6}	3.12×10^{-6}	1.67×10^{-6}
10^{-8}	10^{-2}	3.75×10^{-0}	8.01×10^{-1}	7.44×10^{-4}	3.96×10^{-4}	9.63×10^{-3}	3.57×10^{-3}	4.65×10^{-3}
10^{-6}	0	3.71×10^{-5}	1.05×10^{-5}	1.65×10^{-6}	2.96×10^{-7}	8.82×10^{-6}	2.57×10^{-5}	6.61×10^{-6}
10^{-6}	10^{-2}	4.71×10^{-1}	1.35×10^{-1}	2.12×10^{-3}	9.16×10^{-4}	1.82×10^{-2}	2.40×10^{-2}	1.63×10^{-2}
10^{-4}	0	9.36×10^{-3}	2.20×10^{-2}	3.19×10^{-3}	2.66×10^{-3}	1.01×10^{-1}	4.34×10^{-1}	1.24×10^{-1}
10^{-4}	10^{-2}	1.33×10^{-2}	1.00×10^{-3}	9.33×10^{-4}	9.26×10^{-4}	2.65×10^{-2}	7.46×10^{-2}	3.87×10^{-2}
10^{-2}	0	1.08×10^{-4}	6.42×10^{-6}	1.74×10^{-2}	1.51×10^{-2}	5.31×10^{-1}	3.77×10^{-1}	4.66×10^{-1}
10^{-2}	10^{-2}	3.17×10^{-4}	4.70×10^{-5}	3.20×10^{-3}	2.23×10^{-3}	3.81×10^{-2}	1.77×10^{-1}	1.12×10^{-1}

TABLE V. Error on the recovered parameters for the test case A with and without gradient penalization. The gray colour indicates that the minimiser did not converge.

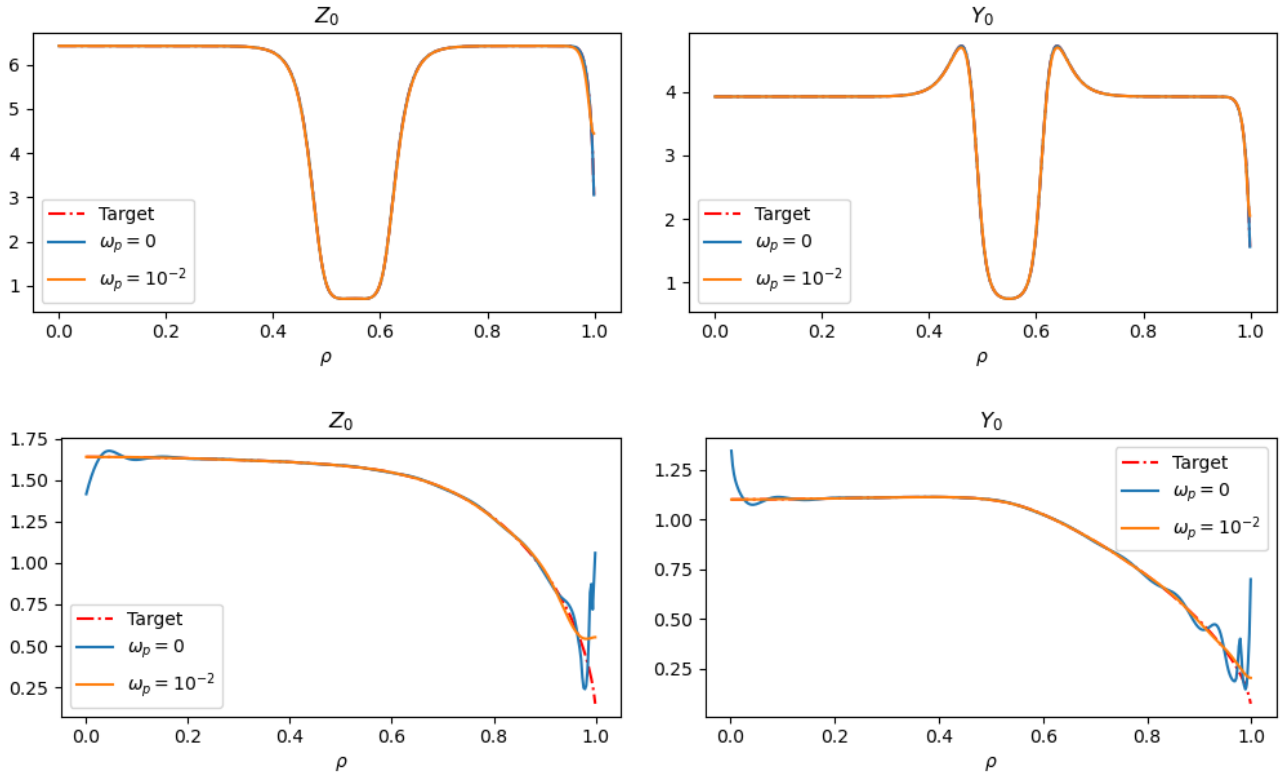

FIG. 22. Identified shapes for Z_0 and Y_0 and their targets for two different values of the target D_{gB}, 10^{-6} (top line) and 10^{-2} (bottom line), of the test case A, on the time interval $[40;50]$, with and without penalisation.

the target Z_0 and Y_0 and of their values retrieved by the minimisation algorithm. For higher D_{gB}^{obj} with the time interval [40;50] the parameters obtained without the penalization are indistinguishable from the target values, whereas with the penalization the points with the higher second spatial derivative are slightly off (typically at the right boundary for both Z_0 and Y_0 and the bumps around the central gap for Y_0). This slight error seems fairly tolerable, and will very likely be negligible with regards to the imprecision of the model and of the measurements. Concerning the highest D_{gB}^{obj}, the penalization seems rather beneficial specifically on the boundaries. In this case it appears quite difficult for the algorithm to obtain a precise shape at the right boundary, so the flattened version retrieved with the penalization is probably the best that could be obtained without spending too much iterations (and the case of $D_{gB}^{obj} = 10^{-4}$ shows that they may actually not converge).

In this case the parameter fitting algorithm behaves well with just a little tuning of the scaling of the parameters. The inclusion of a penalization on the gradient makes the algorithm robust in every tested case despite a loss on the precision of the parameters for lower D_{gB} and a slight increase of the number of iterations, what looks promising for its application to real data (i.e. not generated by the model itself).

B. Test case B :

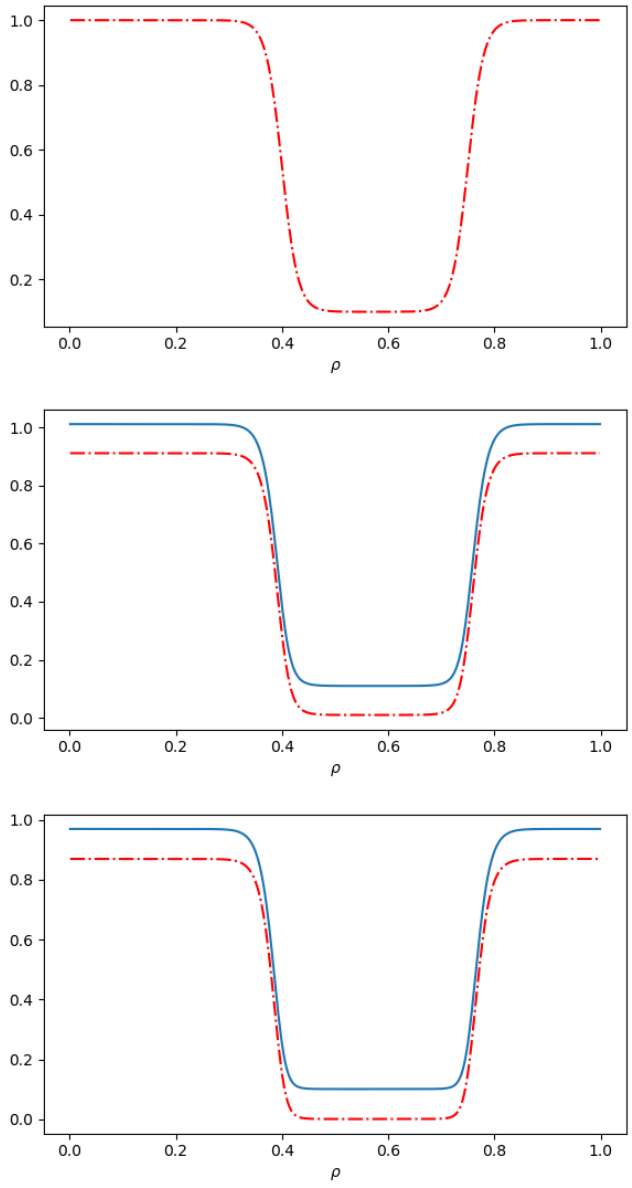

FIG. 23. Profile targets and first guesses for the non constant parameters in the test case B. From top to bottom, $\gamma_Z = \gamma_Y$, Z_0, Y_0.

spreading with variation of γ_Y, fixed u and $D_{gBY} \neq D_{gBZ}$

We now consider as the target of the parameter fitting procedure the No Man's Land configuration of Section IV B 2.

This configuration appears to be notably more demanding than the previous one for the algorithm. Firstly, the large differences in scale of some target parameters make them close to sets of parameters for which the direct solver is unstable, that could prevent the convergence of the algorithm. Secondly, the initial state is the fixed point of the local model (without the diffusion term) which is not far from the fixed point of the whole model. Hence, Z and Y do not move much outside of the region with a lower growth rate, what will give few local

	D_{gBZ}	D_{gBY}	γ_Z	γ_Y
Initial guess	10^{-5}	10^{-5}	1.2	0.7
Target value	5×10^{-3} to 10^{-4}	$2^{-6}\times D_{gBZ}$	NML shape	NML shape

	Z_0	Y_0	K
Initial guess	target+0.1	target+0.1	2×10^{-2}
Target Value	NML shape	NML shape	5×10^{-2}

TABLE VI. Initial guess and target values of parameters for the test case B (the NML shape stands for No Man's Land shape, see Figure 23 for the non constant parameters).

	D_{gBZ}	D_{gBY}	γ_Z	γ_Y	K	Z_0	Y_0
type	exp.	exp.	lin.	lin.	lin.	PQL	PQL
params	1.0	1.0	1.0	1.0	1.0	1.0 & 10.0	1.0 & 10.0

TABLE VII. Scaling function for the different parameters for the test case B. Here, exp., lin. and PQL stand, respectively, for exponential, linear and positive quasi-linear.

information for the computation of the parameters except the local fixed point. Indeed, there is an infinity of sets of values for γ_Z, γ_Y and K which lead to the same local fixed point. Thirdly, we will mostly consider relatively high values for the target D_{gBZ} and D_{gBY}, an issue that alone has proven, in Section V A, to be more challenging for the algorithm.

Accordingly, we could only obtain the convergence of the algorithm with the initial guesses for Z_0, Y_0 quite close to their target values, specifically with very similar shapes. Otherwise the algorithm stops quickly at a state where it cannot find a suitable step along the descent direction, usually because the set of parameters is too close to a region where the model is unstable. The first guesses for the initial values Z_0 and Y_0 are then simply defined as their target values plus 0.1 (see Figure 23). It seems also useful to put back their linear scaling coefficient to 1 (while it was 5 in the previous section), supposing we have a relatively good confidence on those guesses. The quasi linear positive scaling function for Z_0 and Y_0 is indeed necessary here since the target Z_0 and Y_0 are very close to 0. The linear scaling coefficient of D_{gBZ} and D_{gBY} is also set back to 1.0 because otherwise, their values are moving a bit too erratically during the first tens of iterations, eventually reaching values far from the target, from where it is then difficult to come back. Table VII shows the scaling functions and their internal arguments that have been used for each parameter.

Except for Z_0 and Y_0, the first guesses for the parameters are kept similar to those in the previous case, so most of them are constant. Table VI summarizes the targets and initial guesses and Figure 23 shows the profile of non constant parameters.

The penalization on the gradient of the parameters is in this case necessary for the convergence of the algorithm, because the apparition of oscillation close to 0 in Z_0 and Y_0 will quickly push the solver to the limit of its stability. We keep exactly the same procedure as for the preceding case with two runs of the minimiser `m1qn3`, first with a larger penalisation weight ($w_p = 0.01$) and a relaxed stopping condition ($\varepsilon_g^i = 10^{-5}$), and a second one with a lower weight ($w_p^+ = 10^{-4}$) and a more

target D_{gB}	Iterations	Simulations	Final cost
5×10^{-3}	755	804	8.00×10^{-8}
10^{-3}	876	930	3.59×10^{-8}
5×10^{-4}	1078	1139	2.52×10^{-8}
10^{-4}	2204	2373	2.69×10^{-8}

TABLE VIII. Number of iterations and final cost (without penalization) for the test case B over the time interval [0,10].

precise stopping condition $\varepsilon_g = 10^{-7}$. We also introduce for this case the penalisation of negative values on each parameter (see the paragraph on penalization in Section III B). It is useful here because in some instances the penalization of the gradient is stronger than the gradient due to the differences with the data, so that K is drifting into negative values to compensate for the steep slopes of γ_Z and γ_Y around the region with a weaker growth rate. We set the weight of this penalisation to 0.1. It does not have to be changed between the two runs of `m1qn3` because it should not influence the precision of the retrieved parameters (since they should not be negative).

Finally, although the main oscillations of large amplitude in the No Man's Land zone have a period clearly longer than 10, longer time intervals do not seem to reduce the number of iterations overall, so we keep the usual time interval length of 10. However, since we consider a convergent case and the initial conditions are not very flat, we have not try to use initial conditions obtained at later times, at least for the first series of minimisations. The data then spans on the interval $[0,10]$ of the simulations of subsection IV B 2.

Tables VIII and IX show the results of the minimisation for a constant ratio of diffusion weights $\Delta_Y = 2^{-6}$ and different target D_{gB} spanning this time only from 10^{-4} to 5×10^{-3} because they lead to the strongest oscillations of the maximum of $D_{Z,Y}$ (see Figure 17). As it can be seen, the number of iterations increases as the target D_{gB} decreases, and accelerates below 5×10^{-4}. Indeed below this value, Z and Y hardly depart from their initial positions even in the No Man's Land region, limiting the available information for the evaluation of the parameters. This is especially true for K because the saturation term $(-KZ^2)$ is at least an order of magnitude inferior to the other terms at all time for a simulation with the target parameters.

Accordingly the error on the retrieved parameters is the worst for the lowest D_{gB}^{obj} and decreases until a minimum is reached around $D_{gB}^{obj} = 10^{-3}$. For higher D_{gB}^{obj} we may have the same problem of blurring as in the case A but it is also likely that the gradient norm is not reduced enough to reach the stopping threshold before the parameters have been precisely identified. The parameter K generally has a low accuracy, and as shown in Figure 24 the fall of precision is not only at the boundaries but more or less everywhere in the span of the radius. This precision may not seem very satisfying, but mostly shows the low impact of this parameter on the behaviour of the $\kappa-\varepsilon$ model in this case.

The retrieved parameter γ_Y, displayed in Figure 25 for the different sub-cases, shows that the parameters are more difficult to identify around the core boundary ($\rho \to 0$). Indeed,

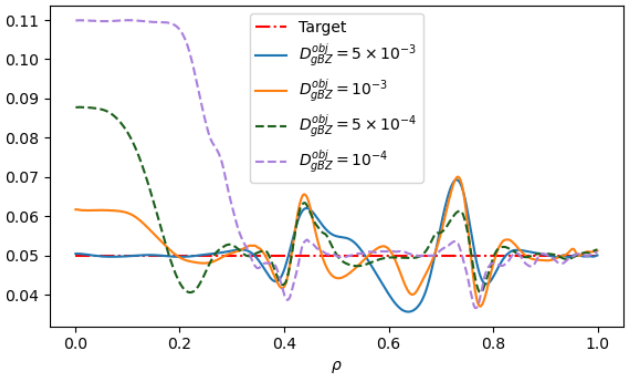

FIG. 24. Final identified parameter K for the case B and different target D_{gB}, using the time interval [0,10].

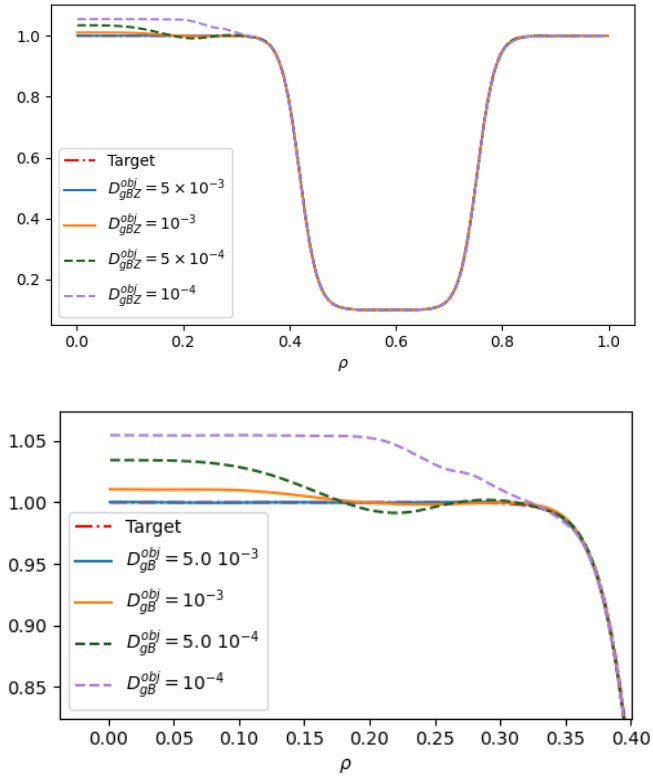

FIG. 25. Final identified parameter γ_Y for the test case B and different target D_{gB}, using the time interval [0,10], and zoom on the core (left) boundary.

while at the other boundary the homogeneous Dirichlet condition bends the initially constant shape, nothing pushes Z and Y away from the local fixed point where they are initialized except a very tiny diffusion because the shape is not exactly constant. It is almost as if only the information of the stationary limit was available there, and due to its dimension, it can be obtained with an infinite set of different parameters. Typically, the shapes of K and γ_Y are very similar at the left boundary because they compensate each other.

D_{gB}^{obj}	D_{gBZ}	D_{gBZ}	γ_Z	γ_Y	K	Z_0	Y_0
5.00×10^{-3}	1.71×10^{-5}	1.53×10^{-4}	5.56×10^{-3}	3.34×10^{-2}	1.23×10^{-1}	1.31×10^{-1}	2.40×10^{-3}
1.00×10^{-3}	8.60×10^{-5}	1.74×10^{-4}	4.80×10^{-3}	8.28×10^{-3}	1.37×10^{-1}	5.80×10^{-4}	3.45×10^{-4}
5.00×10^{-4}	3.06×10^{-4}	2.61×10^{-3}	1.19×10^{-2}	1.22×10^{-2}	2.70×10^{-1}	2.96×10^{-4}	2.29×10^{-4}
1.00×10^{-4}	4.09×10^{-4}	9.99×10^{-3}	2.67×10^{-2}	2.67×10^{-2}	5.89×10^{-1}	2.29×10^{-4}	3.08×10^{-4}

TABLE IX. Error on the recovered parameters for the test case B over the time interval [0,10].

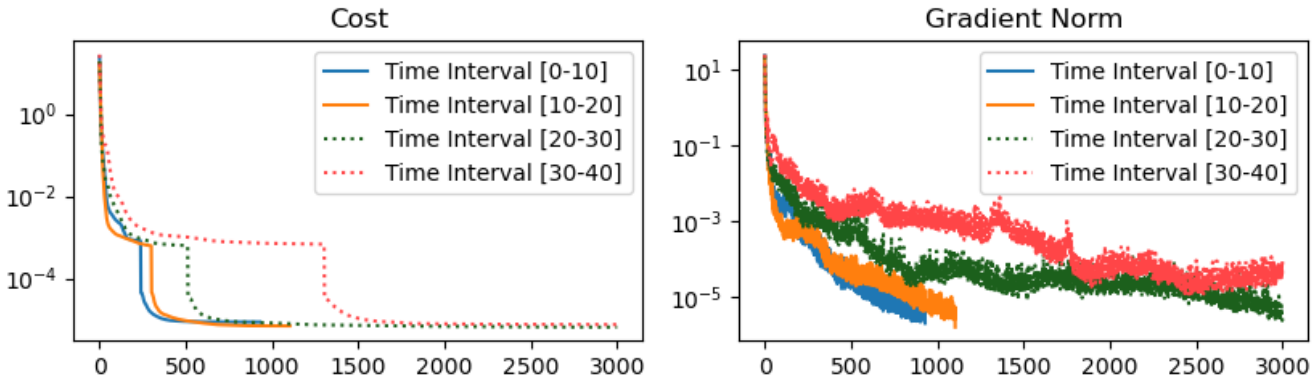

FIG. 26. Evolution of the cost and the norm of its gradient for different time intervals, for a target D_{gBZ} of 10^{-3} of the test case B. The step in the cost reflects the sudden decrease of the penalisation of the gradient as its weight is changed passing from the first minimisation to the second.

Lastly, it appears that the number of iterations necessary for the convergence of the algorithm greatly increases with the lower bound of the time interval on which the cost is computed as seen in Figure 26: the limit of 3000 iterations is reached starting from the time interval [20; 30]. This could be explained by the fact that in this case the model naturally evolves towards a stationary limit, so should progressively have less and less movement. And while the model still shows visible oscillations at a time of 400, this is only true for the radii inside the region of lower growth rate. Hence the model is very slow to identify the parameters outside of the lower growth rate region, because the model mostly does not move at those points.

VI. CONCLUSION

The paper goes further in the analysis of the dynamics of the reduced $\kappa-\varepsilon$ model, and shows the potential of a data assimilation technique to estimate the inherent free parameters. It is of primary importance to be able to accurately estimate the free parameters of this model in order to increase the reliability of fluid plasma simulations and the predictive capability of the codes.

Assuming a Fick's law for the transverse turbulent transport, this model allows to self-consistently estimate the cross-field fluxes in the edge and SOL regions of diverted plasma. The nonlinear dependence of D in κ and ε, both quantities being estimated from two additional transport equations, allows to introduce some nonlocality in the transport model and to show the existence of parameter ranges where the turbulence is spread. This feature of the model is important since stepping to flux driven simulations, one can then expect that spreading will be more effective. Indeed, the modified transport governed by the κ-ε will also modify the instability growth rate via the evolution of the gradients, hence generating a higher dimension phase space where non-linear effects can drive novel transport phenomena and in particular enhanced spreading properties. The asymptotic behaviour of the system showed also the existence of limit cycles with amplitude and frequencies depending on the the growth rate ratio for κ and ε. Such a behaviour has an impact on the parameter identification since the result of the simulation for the current situation may have different local periods of oscillation from the ones of the target data.

Regarding data assimilation, the variationnal approach has been validated for the estimation of several model parameters on two cases corresponding to limit cycles when varying $u=\gamma_\varepsilon/\gamma_\kappa$ the growth rate ratio for κ and ε or only γ_κ. In the first case, the parameter fitting algorithm behaves well: the cost function and its gradient are significantly decreased, and the parameters are very well identified. Adding a penalization term (that makes the gradient more regular) leads to a robust algorithm with very good results in all cases. In the

second case, the algorithm works again relatively well, but it usually requires more iterations to converge, and the parameters are not as precisely identified as in the first experiment. It is still satisfying to see that the algorithm can converge on an arguably more complex case if the first guesses for the initial states are not too far from their target values. However, the problems of convergence due to the lack of information at some points seem to indicate that, in this case, the identification of parameters may not be able to efficiently deal with rarefied data on the radius.

In a future work, we plan to study in detail the sensitivity of the parameter identification procedure with respect to data density/sparsity and to noise on data. Then, experimental data (or data extracted from model simulations with a more complex model) could be considered.

ACKNOWLEDGEMENTS

This work has been supported by the French National Research Agency grant SISTEM (ANR-19-CE46-0005-03). This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200 - EUROfusion). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

[1] A. Loarte, B. Lipschultz, A. Kukushkin, G. Matthews, P. Stangeby, N. Asakura, G. Counsell, G. Federici, A. Kallenbach, K. Krieger, and et al., "Power and particle control," Nuclear Fusion **47**, S203 (2007).

[2] S. W. et al., "The new solps-iter code package," J. Nucl. Mater. **463**, 480–484 (2015).

[3] H. Bufferand, G. Ciraolo, Y. Marandet, J. Bucalossi, P. Ghendrih, J. Gunn, N. Mellet, P. Tamain, R. Leybros, N. Fedorczak, F. Schwander, and E. Serre, "Numerical modeling for divertor design of the west device with a focus on plasma wall interactions," Nuclear Fusion **55**, 053025 (2015).

[4] S. Pope, *Turbulent flows* (Cambridge University Press, 2000).

[5] Naulin, "Turbulent transport and the plasma edge," J. of Nucl. Mat. **363-365**, 24–31 (2007).

[6] L. Aho-Mantila, M. Wischmeier, H. Müller, S. Potzel, D. Coster, X. Bonnin, G. Conway, and the ASDEX Upgrade Team, "Outer divertor of asdex upgrade in low-density l-mode discharges in forward and reversed magnetic field: I. comparison between measured plasma conditions and solps5.0 code calculations," Nuclear Fusion **52**, 103006 (2012).

[7] F. Hasenbeck, D. Reiser, P. Ghendrih, Y. Marandet, P. Tamain, A. Moller, and D. Reiter, "Multiscale modeling approach for radial particle transport in large-scale simulations of the tokamak plasma edge," Procedia Computer Science **52**, 1128–1137 (2015).

[8] Y. Nishimura, K. Borrass, D. Coster, and B. Scott, "Effects of resistive drift wave turbulence on tokamak edge transport," Contr. Plasma Phys. **44**, 194–199 (2004).

[9] K. Miki, P. H. Diamond, L. Schmitz, D. C. McDonald, T. Estrada, O. D. Gurcan, and G. R. Tynan, "Spatio-temporal evolution of the H⟶L back transition," Phys. of Plasma **20**, 062304 (2013).

[10] G. R. Tynan, I. Cziegler, P. H. Diamond, M. Malkov, A. Hubbard, J. W. Hughes, J. L. Terry, and J. H. Irby, "Recent progress towards a physics-based understanding of the h-mode transition," Plasma Phys. and Control. Fusion **58** (2016).

[11] B. Launder and D. Spalding, "The numerical computation of turbulent flows," Comput. Methods Appl. Mech. Eng. **3** (1974).

[12] P. W. Terry, A. F. Almagri, G. Fiksel, C. B. Forest, D. R. Hatch, F. Jenko, M. D. Nornberg, S. C. Prager, K. Rahbarnia, Y. Ren, and J. S. Sarff, "Dissipation range turbulent cascades in plasmas," Phys. of Plasma **19**, 055906 (2012).

[13] S. Baschetti, H. Bufferand, G. Ciraolo, N. Fedorczak, P. Ghendrih, E. Serre, and P. Tamain, "Optimization of turbulence reduced model free parameters based on l-mode experiments and 2d transport simulations," Contr. Plasma Phys. **58**, 511–517 (2018).

[14] S. Baschetti, H. Bufferand, G. Ciraolo, P. Ghendrih, E. Serre, P. Tamain, and the WEST team, "Self-consistent cross-field transport model for core and edge plasma transport," Nuclear Fusion **61**, 106020 (2021).

[15] A. F. Bennett, *Inverse Modeling of the Ocean and Atmosphere* (Cambridge University Press, Cambridge, 2002).

[16] E. Kalnay, *Atmospheric modeling, data assimilation and predictability* (Cambridge University Press, 2003).

[17] F.-X. L. Dimet and O. Talagrand, "Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects," Tellus **38A**, 97–110 (1986).

[18] K. Miki, P. Diamond, O. Gürcan, G. Tynan, T. Estrada, L. Schmitz, and G. Xu, "Spatio-temporal evolution of the l $\rightarrow$ i $\rightarrow$ h transition," Phys. of Plasma **19**, 092306 (2012).

[19] R. Balescu, I. Petrisor, and M. Negrea, "Anisotropic electrostatic turbulence and zonal flow generation," Plasma Physics and Controlled Fusion **47**, 2145–2159 (2005).

[20] G. Qin and A. Shalchi, "The role of the kubo number in two-component turbulence," Physics of Plasmas **20**, 092302 (2013), https://doi.org/10.1063/1.4821026.

[21] L. Hascoët and V. Pascual, "The tapenade automatic differentiation tool: principles, model, and specification," Research Report RR-7957 (INRIA, 2012).

[22] J.-C. Gilbert and C. Lemaréchal, *The module M1QN3* (2009), accessed 2021-12-01.

[23] J.-C. Gilbert and C. Lemaréchal, "Some numerical experiments with variable-storage quasi-newton algorithms," Mathematical Programming **45**, 407–435 (1989).

[24] F. Devernay, "C/c++ minpack," `http://devernay.free.fr/hacks/cminpack/` (2007).

[25] P.-A. Raviart and J.-M. Thomas, *Introduction to the numerical analysis of the PDEs* (Masson, 1983).

[26] C. Bourdelle, J. Citrin, B. Baiocchi, A. Casati, P. Cottier, X. Garbet, and F. Imbeaux, "Core turbulent transport in tokamak plasmas: bridging theory and experiment with qualikiz," Plasma Phys. Control. Fusion **58**, 014036 (2015).