Gyrokinetic simulations of magnetic fusion plasmas

Tutorial 3

Virginie Grandgirard

CEA/DSM/IRFM, Association Euratom-CEA, Cadarache, 13108 St Paul-lez-Durance, France.

email: virginie.grandgirard@cea.fr

Acknowledgements: Yanick Sarazin

Gyrokinetic codes

Physical complexity in the gyrokinetic codes

- δf vs full-f

  $- \xrightarrow{complex} +$

  - No scale separation between equilibrium and perturbation
- Local ("flux-tube") approximation vs. global model
  - Covering just a local or the whole physical domain
- Adiabatic vs. kinetic electrons
  - Taking the full kinetics of all species into account
- Electrostatic vs. electromagnetic model
  - Taking self generated currents and Ampère's law into account
- Collisionless vs. collisional plasma
  - Taking collisional effects into account (neoclassical theory)
- Fixed gradient or flux-driven boundary conditions

None of the codes covers all physical aspects

Physical complexity in the gyrokinetic codes

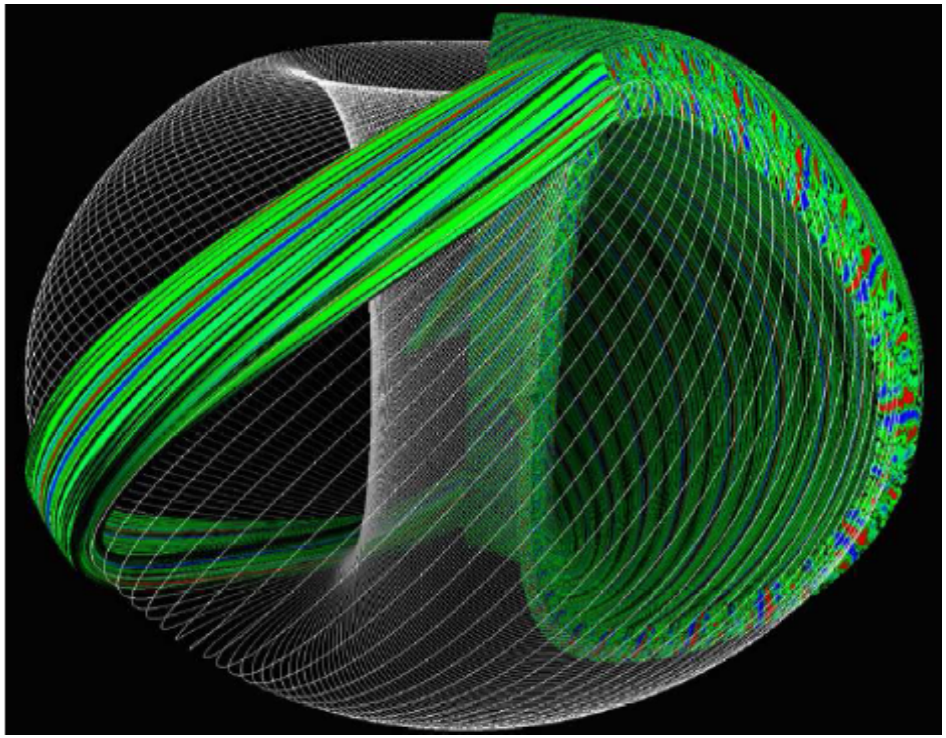
- δf vs full-f

  $- \xrightarrow{complex} +$

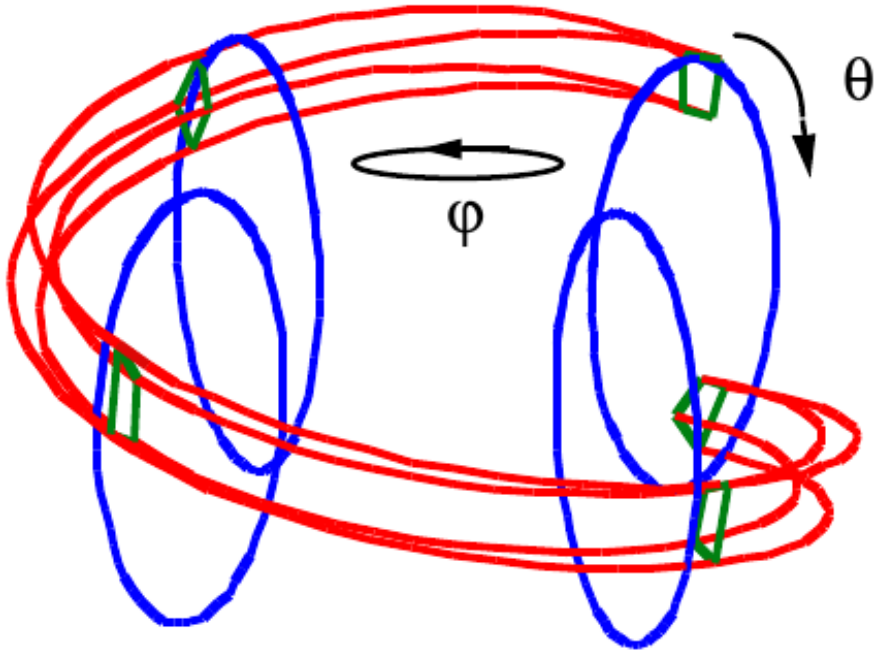
  - No scale separation between equilibrium and perturbation
- Local ("flux-tube") approximation vs. global model
  - Covering just a local or the whole physical domain
- Adiabatic vs. kinetic electrons
  - Taking the full kinetics of all species into account
- Electrostatic vs. electromagnetic model
  - Taking self generated currents and Ampère's law into account
- Collisionless vs. collisional plasma
  - Taking collisional effects into account (neoclassical theory)
- Fixed gradient or flux-driven boundary conditions

GYSELA: 4 complexities on 6

Flux-tube geometry



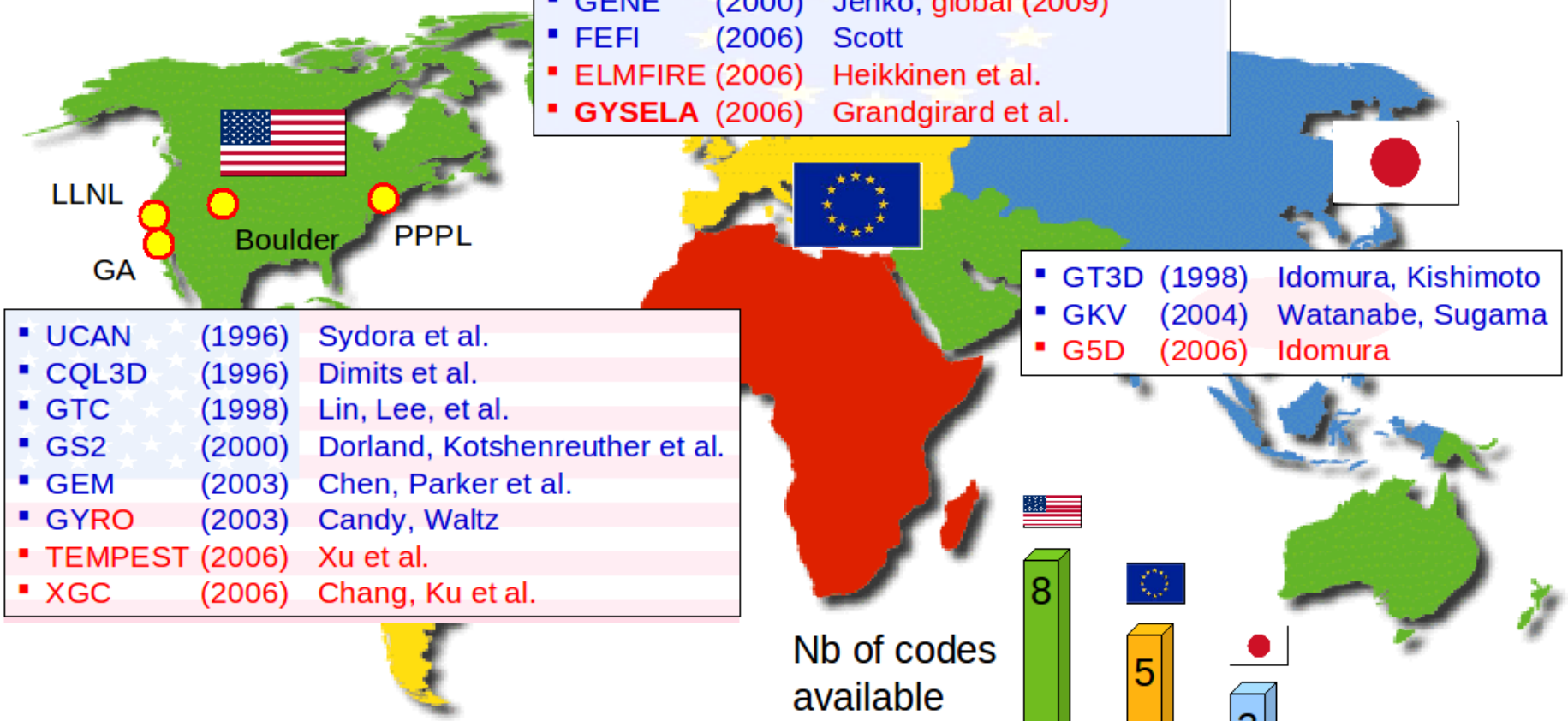
GS2 code



Existing gyrokinetic codes

Local / Global codes

- ORB5 (1999) Tran, Brünner, global (2007)
- GENE (2000) Jenko, global (2009)
- FEFI (2006) Scott
- ELMFIRE (2006) Heikkinen et al.
- **GYSELA** (2006) Grandgirard et al.



- UCAN (1996) Sydora et al.
- CQL3D (1996) Dimits et al.
- GTC (1998) Lin, Lee, et al.
- GS2 (2000) Dorland, Kotshenreuther et al.
- GEM (2003) Chen, Parker et al.
- GYRO (2003) Candy, Waltz
- TEMPEST (2006) Xu et al.
- XGC (2006) Chang, Ku et al.

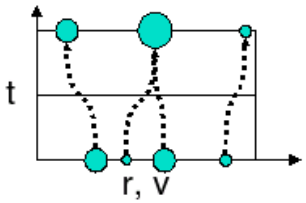
- GT3D (1998) Idomura, Kishimoto
- GKV (2004) Watanabe, Sugama
- G5D (2006) Idomura

Gyrokinetic codes require state-of-the-art HPC

- Three existing numerical approaches

① Particle-in-Cell (PIC)

follow trajectories



☹ numerical noise

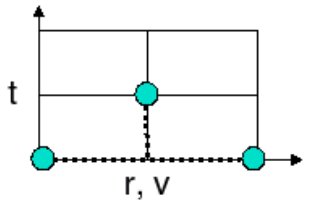
★ optimized loading

② Eulerian

fixed grid

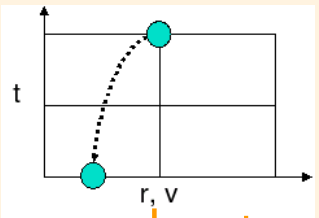
☹ dissipation

★ high order scheme



③ Semi-Lagrangian scheme

- fixed grid
- calculate trajectories backwards
- interpolation



weak noise, moderate dissipation

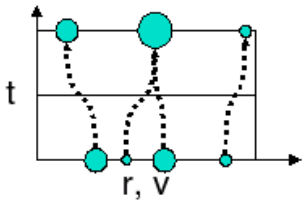
GYSELA : GYrokinetic SEmi-LAgrangian code

Gyrokinetic codes require state-of-the-art HPC

- Three existing numerical approaches

① Particle-in-Cell (PIC)

follow trajectories



☹ numerical noise

★ optimized loading

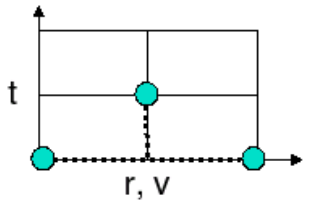
// Monte-Carlo

② Eulerian

fixed grid

☹ dissipation

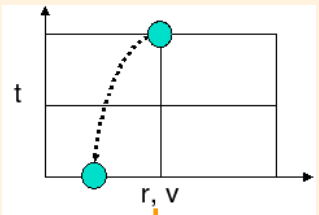
★ high order scheme



// Domain decomposition

③ Semi-Lagrangian scheme

- fixed grid // Domain decomposition
- calculate trajectories backwards
- interpolation ☹ Non-local



weak noise, moderate dissipation

GYSELA: more accurate but more difficult to parallelize

Parallelisation of a Semi-Lagrangian method

Advantage *(due to the eulerian aspect)* :

- fixed grid ➠ perfect load balancing

Drawback *(due to interpolation)* :

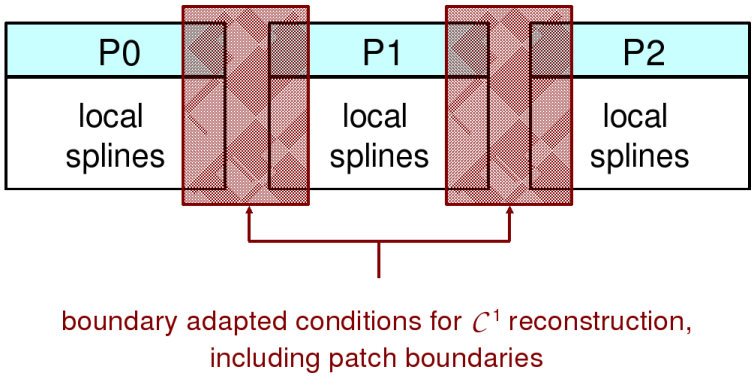
- Several choices for the interpolation
- But we use cubic splines interpolation :
  - ☺ Good compromise between accuracy and simplicity
  - ☹ Loss of locality
    (value of f on one grid point requires f over the whole grid)

➠ **Not possible to use a simple domain decomposition**

New approach : Local cubic splines

A new numerical tool has been developed

➠ Hermite Spline interpolation on patches *[Latu-Crouseilles '07]*



- Computational domain decomposed in subdomains
- Definition of local splines on each subdomains with Hermite boundary conditions
- Derivatives are defined so that they match as closely as possible those of global splines

What are the ingredients of a GK code ?

1. One 3D quasi-neutrality solver
2. One gyroaverage operator
3. One 5D Vlasov solver

Numerical treatment of the quasi-neutrality equation

- The treatment of the quasi-neutrality equation is almost the same for all the codes
- Projection in Fourier space in the periodic directions
- Finite differences or finite elements in 1D or 2D to solve the Laplacian

Numerical treatment of the gyroaverage

- In Fourier space, the gyro-average reduces to the multiplication by the Bessel function of argument $k_\perp \rho_s$.
- This operation is straightforward in simple geometry with periodic boundary conditions, such as in local codes.
- Conversely, in the case of global codes, the use of Fourier transform is not applicable for two main reasons:
  - (i) radial boundary conditions are non periodic and
  - (ii) the radial dependence of the Larmor radius has to be accounted for.
- Several approaches have been developed to overcome this difficulty.

Padé approximation

- One of those consists in approximating the Bessel function with Padé expansion

$$J_{\text{Padé}}(k_\perp \rho_s) = 1/\left[1 + (k_\perp \rho_s)^2/4\right]$$

e.g. see *[Y. Sarazin, PPCF (2005)]*.

- Using the equivalence $i\vec{k}_\perp \leftrightarrow \nabla_\perp$, the gyroaverage operation of any g leads to the equation

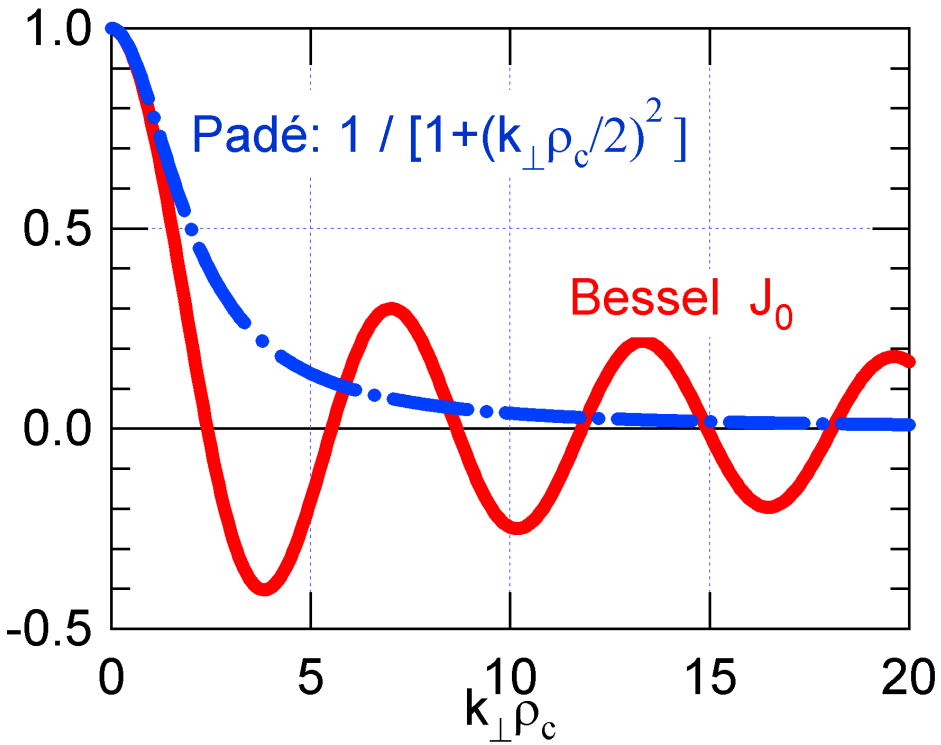
$$\left[1 - (\rho_c^2/4)\nabla^2\right] \bar{g}(r,\theta,\varphi) = g(r,\theta,\varphi)$$

where we recall that $\nabla_\perp^2 = \frac{\partial^2}{\partial r^2} + (1/r^2)\frac{\partial^2}{\partial \theta^2}$

- Such a Padé representation then requires the inversion of the Laplacian operator $\nabla_\perp^2$ in real space.

Correct limit in large wavelengths limit

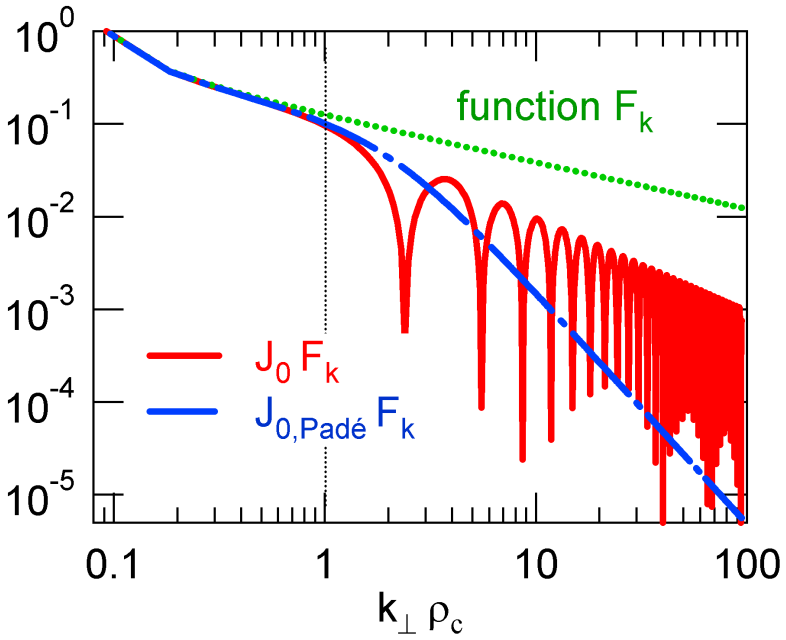
- This approximation gives the correct limit in the large wavelengths limit $k_\perp \rho_c \ll 1$, while keeping $J_{\text{Padé}}$ finite in the opposite limit $k_\perp \rho_s \to \infty$



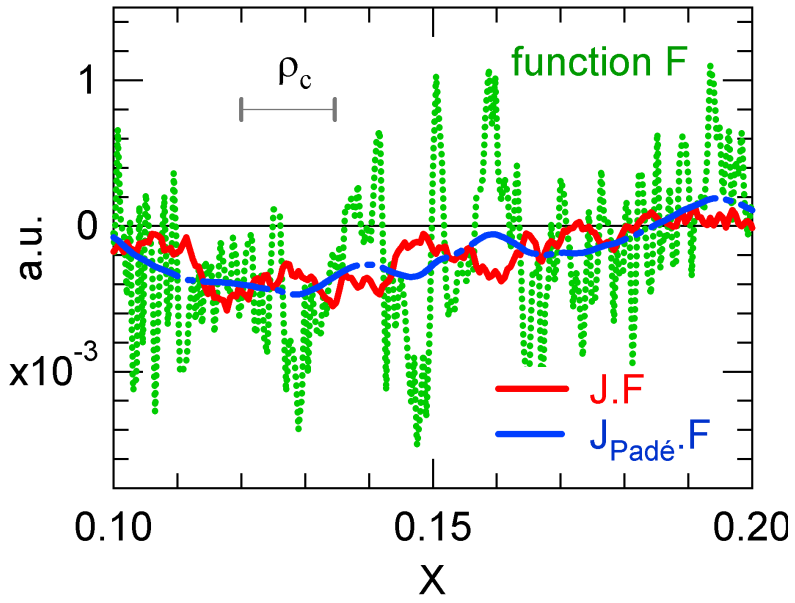
Padé approximation drawback

- The drawback is an over-damping of small scales: in the limit of large arguments $x \longrightarrow \infty$,

$$J_{\text{Padé}}(x) \longrightarrow 4/x^2 \quad \text{whereas} \quad J_0 \longrightarrow (2/\pi x)^{1/2} \cos(x - \pi/4)$$



Fourier space



real space

Quadrature formula

- Another possibility, for this gyro-averaging process, is to use a quadrature formula.
- The integral over the gyro-ring is usually approximated by a sum over four points on the gyro-ring *[Lee, Phys. Fluid (1983)]*.
- This is rigorously equivalent to considering the Taylor expansion of the Bessel function at order two in the small argument limit, namely $J_0(k_\perp \rho_s) \simeq 1 - (k_\perp \rho_s)^2/4$, and to computing the transverse Laplacian at second order using finite differences

More sophisticated quadrature formula

- This method has been extended to achieve accuracy for large Larmor radius *[R. Hatzky, PoP (2002)*, i.e the number of points (starting with four) is linearly increased with the gyro-radius to guarantee the same number of points per arc-length on the gyro-ring.
- In this approach, the points that are equidistantly distributed over the ring are rotated for each particle (or marker) by a random angle calculated every time step
- Technique used in GT5D code (see *[Idomura, Nuc. Fus. (2003)]*) and ORB5 code (see *[Jolliet, Comp. Phys. Comm. (2007)*)

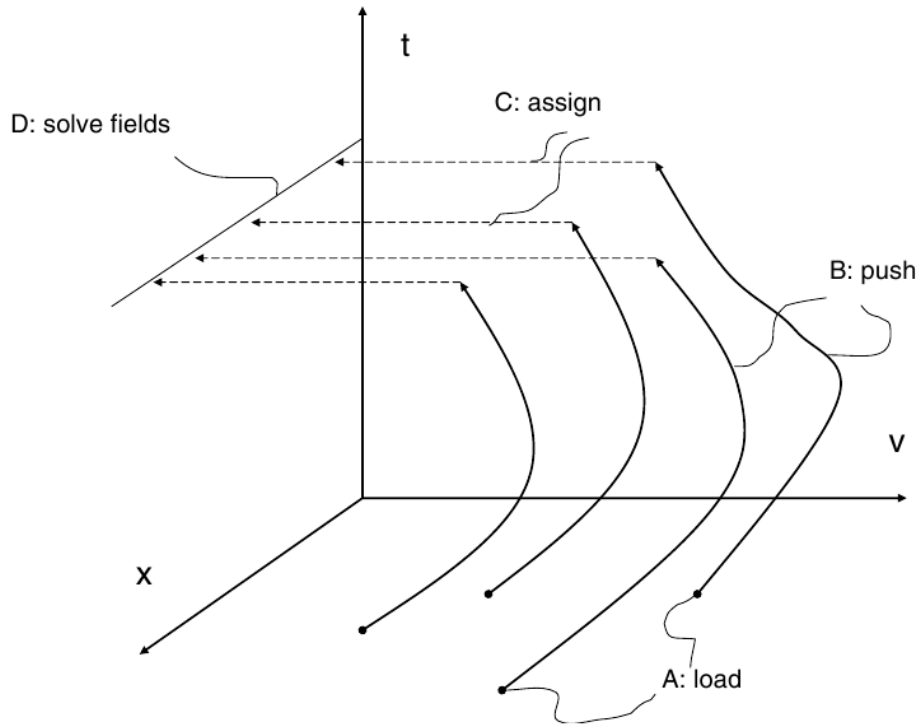
For more details on numerical gyroaverage treatments
$\Rightarrow$ N. Crouseilles, C. Negulescu, M. Mehrenberger

PIC approach

PIC approach in plasma study

- Lagrangian-PIC approach replaces the solution of the partial differential Vlasov equation by the solution of the ODE's of motion of macro-particles.
- Each macro-particles represents a large number of the plasma particles
- In the context of plasma study, the PIC approach is divided into two distinct steps
  - Step 1: Calculating the self-consistent fields generated by a given distribution of computational particles in a multidimensional phase space
  - Step 2: Following the particle orbits (characteristics of Vlasov equation) in these fields

PIC approach



In Lagrangian-PIC methods, marker initial positions are loaded pseudo-randomly in phase space (A). Markers are evolved along their orbits (B). Charge and current perturbations are assigned (projected) to real space (C). Field equations are solved (D), e.g. on a fixed grid in real space. (figure from *[Idomura, CR (2006)]*)

PIC approach ➠ the simplest approach

- The advantage of PIC codes is their simplicity, robustness and scalability.
- The first method used in gyrokinetic theory
- Lots of gyrokinetic codes are based on this method (list not exhaustive)
  - Parker's code *[Parker, PFI (1993)]*, Sydora's code *[Sydora, PPCF (1996)]*, PG3EQ *[Dimits, PRL (1996)]*, GTC *[Lin, Science (1998)]*, ELMFIRE *[Heikkinen, JOCP (2001)]*, GT3D *[Idomura, NF (2003)*, ORB5 *[Bottino, PoP (2007), [Jolliet CPC (2007)]*, GTS *[WangWX, PoP (2007)]*

PIC drawback = noise (1/2)

- Their disadvantage is the numerical noise –caused by the technically limited number of macro-particles– which can cause numerical collisions and artificial dissipation
- Where does this noise come from ?
  - The solution of the dynamical equations in the second step introduces some error and noise
  - But what is called the noise in particle simulations is predominantly associated with the first step, where low-order moments of the distribution function are calculated to find the source terms for Poisson's or Ampere's equations.

⇛ This noise is essentially due to the error introduced when evaluating the moments using a relatively small number of points in phase space, determined by computational particle positions

PIC drawback = noise (2/2)

- The main problem for non-linear gyrokinetic simulations is that the noise level can accumulate indefinitely (see *[Nevins, PoP (2005)]*) and that even small errors in the evaluation of theses moments can cause a systematic corruption of the simulation in a relatively short period of time.
- The research of solutions to reduce this numerical noise in PIC code is right from the start a subject of great importance and lots of progresses have been performed in this domain since five years.
- In particular, lots of improvements have been performed in the ORB5 gyrokinetic PIC-code

PIC algorithm ➠ Monte-Carlo integration

- Mathematically, Aydemir [*Aydemir, PoP (1994)*] has point the fact that the Lagrange-PIC algorithm can be viewed as a statistical method to obtain estimates of the moments of the distribution function, via *Monte Carlo integration*
- For the following the term "Monte-Carlo" will refer to estimation of multidimensional integrals using statistical sampling techniques.
- In general, particular form of the integrand and how it is sampled in the volume of interest determine the accuracy of the estimates.
- Since the 1950s, the Monte Carlo community has developed a number of techniques that try to minimize the error in the estimates and increase the efficiency of the calculations.

- For the discussion let consider a general integral of the form:

$$I(\Upsilon) = \int_V \Upsilon(\mathbf{Z}) f(\mathbf{Z}) \, \mathrm{d}\tau \tag{1}$$

where $\mathrm{d}\tau = \mathcal{J} d\mathbf{x}\, d\mathbf{v}$ is the volume element, $\Upsilon(\mathbf{Z})$ is a general function of the phase-space coordinates $\mathbf{Z} = (\mathbf{x}, \mathbf{z})$ and $f(\mathbf{Z}, t)$ is the distribution function of some population of Ns particles, i.e

$$\int_V f(\mathbf{Z}) \, \mathrm{d}\tau = N_s$$

- For instance, $I(\Upsilon)$ would be the number density in configuration space if $\Upsilon = 1$, and the integral is over the velocity space.
- Let treat $\mathbf{Z}$ as a continuous random variable with a probability density function (PDF) $p(\mathbf{Z})$ in the phase-space volume V, the sampling distribution satisfying,

$$\int_V p(\mathbf{Z}) \, \mathrm{d}\tau = 1 \tag{2}$$

Basic principle of Monte-Carlo methods (1/2)

- Then the basic principle of Monte-Carlo methods is to see the previous equation (1) ($I(\Upsilon) = \int_V \Upsilon(\mathbf{Z}) f(\mathbf{Z}) \, \mathrm{d}\tau$) as

$$I(\Upsilon) = \mathbb{E}_p(g(\mathbf{Z})) = \int_V g(\mathbf{Z}) f(\mathbf{Z}) \, \mathrm{d}\tau \tag{3}$$

where $\mathbb{E}_p(g)$ is the expected value of the random variable

$$g \equiv (\Upsilon(\mathbf{Z}) f(\mathbf{Z}))/p(\mathbf{Z}) \tag{4}$$

under the probability density $p(\mathbf{Z})$ $\quad$ ($\int_V p(\mathbf{Z}) \, \mathrm{d}\tau = 1$)

Basic principle of Monte-Carlo methods (2/2)

- Let also define the variance of g by

$$\mathcal{V}(g) = \sigma_g^2 = \int_V (g - \mathbb{E}_p(g))^2 p(\mathbf{Z})\, \mathrm{d}\tau \tag{5}$$

- The idea is to produce an independent random sample $(\mathbf{Z}_1, \mathbf{Z}_2, \cdots, \mathbf{Z}_N)$ for the random variable $\mathbf{Z}$ of probability $p(\mathbf{Z})$

- and to calculate a new estimate (called Monte-Carlo estimate) in function of this sampling.

- The law of large number (first fundamental theory of probability) suggests to generate this estimate with the empiric mean

$$\tilde{g}_N = \frac{1}{N}\sum_{j=1}^{N} g(\mathbf{Z}_j) \tag{6}$$

- It is to notice that N the number of markers is limited by the computational power and therefore $N \ll N_s$.
- Then, let S_N the random variable be defined such that $\mathbb{E}_p(S_N) = 0$ and $\sigma(S_N) = 1$ by

$$S_N \equiv \frac{\tilde{g}_N - \mathbb{E}_p(\tilde{g}_N)}{\sigma_g/\sqrt{N}}$$

where the unbiased estimate $\tilde{g}_N$ (i.e $\mathbb{E}_p(\tilde{g}_N) - \mathbb{E}_p(g(\mathbf{Z})) = 0$) is defined by Eq. (6), the expected value $\mathbb{E}_p$ by Eq. (3) and the square root of the variance σ_g by Eq. (5)

The central limit theorem

- The central limit theorem (second fundamental theorem of probability) states that S_N will converge in distribution to the standard normal distribution $\mathbb{N}(0;1)$ as N approaches infinity.

- Convergence in distribution means that if $\Phi(z)$ is the cumulative distribution function of $\mathbb{N}(0;1)$, i.e

$$\Phi(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t}{2}\right) \, \mathrm{d}t = \text{error function}$$

then for every real number z, we have

$$\lim_{N\to\infty} P(S_N \leq z) = \Phi(z)$$

Confidence interval

- Therefore it is possible to define a confidence interval capable to indicate the reliability of the estimate $\tilde{g}_N$ compare to the moment integral $I(\Upsilon)$.
- Let e_N be this error, then for a confident level $(1-\alpha)$ $(\alpha \in \mathbb{R})$,

$$|e_N| \leq z_{1-\alpha/2} \frac{\sigma_g}{\sqrt{N}} \tag{7}$$

where the real $z_{1-\alpha/2}$ is the $(1-\alpha)$-th percentile of the distribution.

Example of confidence interval

- Let for instance take $1-\alpha = 0.95$, then

$$P(-z_{1-\alpha/2} \leq S_N \leq z_{1-\alpha/2}) = 1-\alpha = 0.95$$

where $z_{1-\alpha/2}$ follows from the cumulative distribution function:

$$\Phi(z_{1-\alpha/2}) = P(S_N \leq z_{1-\alpha/2}) = 1 - \frac{\alpha}{2} = 0.975$$

$$z_{1-\alpha/2} = \Phi^{-1}(\Phi(z_{1-\alpha/2})) = \Phi^{-1}(0.975) = 1.96$$

- This can be interpreted as, there is 95% of probability to find a confidence interval in which the error between the estimate and the moment integral will be such that

$$|e_n| \leq 1.96 \frac{\sigma_g}{\sqrt{N}}$$

Monte-Carlo estimate for the moment integral

- To summarize, $\tilde{g}_N$ is an *unbiased* (i.e $\mathbb{E}_p(\tilde{g}_N) - \mathbb{E}_p(g(\mathbf{Z})) = 0$) and consistent (i.e $\mathcal{V}(\tilde{g}_N) = \sigma_g^2/N \to 0$ as $N \to \infty$) estimate for the expected value of g, with a standard error $\epsilon \simeq \sigma_g/\sqrt{N}$, where σ_g is the standard deviation defined by Eq. (5).

- Finally, a valid Monte-Carlo estimate for the general moment integral $I(\Upsilon)$ (Eq. 1) is therefore given by

$$I(\Upsilon) = \frac{1}{N}\sum_{j=1}^{N}\frac{\Upsilon(\mathbf{Z}_j)f(\mathbf{Z}_j)}{p(\mathbf{Z}_j)} + \epsilon \quad \text{with} \quad \epsilon \simeq \frac{\sigma_g}{\sqrt{N}} \tag{8}$$

Klimontovitch density

- Besides, let $f_K(\mathbf{Z})$ be the Klimontovitch density

$$f_K(\mathbf{Z}) = \frac{1}{\mathcal{J}} \sum_{j=1}^{N} w_j \delta(\mathbf{Z} - \mathbf{Z}_j) \quad \text{with the weight} \quad w_j = \frac{1}{N} \frac{f(\mathbf{Z}_j)}{p(\mathbf{Z}_j)} \tag{9}$$

- Then it is trivial to see that for any volume element Ω in V moments $I(\Upsilon)$ of f can be expressed as

$$\int_\Omega \Upsilon(\mathbf{Z}) f(\mathbf{Z}) \, \mathrm{d}\tau = \int_\Omega \Upsilon(\mathbf{Z}) f_K(\mathbf{Z}) \, \mathrm{d}\tau + \epsilon, \quad \epsilon \simeq \frac{\sigma_g}{\sqrt{N}} \tag{10}$$

Variance reduction techniques

- Let remark that, in practice, σ_g is unknown and must be estimated.
- One possibility is to use the discrete variance as

$$\sigma_g^2 \simeq \frac{1}{N}\sum_{j=1}^{N}(g(\mathbf{Z}_j) - \tilde{g}_N)^2$$

- Several methods, called variance reduction techniques, allow to improve the accuracy or to reduce the computation time by replacing $g(\mathbf{Z})$ by another random variable.
- Two of them are particularly widespread in plasma particle simulations,
  - the importance sampling
  - the control variates

Importance sampling (1/2)

- The main idea of importance sampling is not to use an uniform marker probability as done in the simplest MC method with

$$p(\mathbf{Z}) = \frac{1}{V}, \quad w_j = \frac{V}{N} f(\mathbf{Z}_j), \quad g = \Upsilon(\mathbf{Z}) f(\mathbf{Z}) V$$

but a non-uniform marker probability proportional to the distribution function:

$$p(\mathbf{Z}) = \frac{1}{N_s} f(\mathbf{Z}), \quad w_j = \frac{N_s}{N} \quad \text{and} \quad g = N_s \Upsilon(\mathbf{Z})$$

➠ to sample more frequently the most "important" regions of the phase-space.

- Remark: In this case, Lagrangian markers are called "macro-particles", each representing N_s/N physical particles.

Importance sampling (2/2)

- One first advantage compare to the simple Monte-Carlo method is that there is no information storage required for the weights, because they are the same for each marker

$$w_j = N_s/N$$

- The second and most important point is that this choice reduce the variation in g because it only comes from the function $\Upsilon(\mathbf{Z})$ since

$$f/p = \text{const}$$

- For these two advantages this importance sampling method is now the basis of lots of PIC simulations in plasma turbulence.
- Hatzky et al. *[Hatzky, PoP (2002)]* have applied successfully such kind of "optimized loading" scheme.

Control variates ➠ The δf method (1/3)

- Control variates method is another intuitively obvious approach that tries to reduce the variance in $I(\Upsilon)$
  - by replacing as much as possible the Monte Carlo estimate by analytic or numerical calculations that are more accurate

- Assume that there exists a function f_0, formally called the control variate, such that
  (i) moments of f_0 can be found easily and preferentially analytically, and
  (ii) at all times, the physical distribution function $f(\mathbf{Z})$ remains close to $f_0(\mathbf{Z})$ in the sense

$$\|f - f_0\|/\|f\| \ll 1$$

  where $\|\cdot\|$ is some arbitrary norm.

Control variates ➠ The δf method (2/3)

- Then the error in the estimate $I(\Upsilon)$, can be reduced by rewriting the integral in the form

$$I(\Upsilon) = \int_V \Upsilon(\mathbf{Z}) f_0(\mathbf{Z})\, \mathrm{d}\tau + \int_V \Upsilon(\mathbf{Z}) \delta f\, \mathrm{d}\tau$$

where

$$\delta f = f(\mathbf{Z}) - f_0(\mathbf{Z})$$

and applying a Monte Carlo technique only to the second integral.

Control variates ➠ The δf method (3/3)

- Therefore, using the same technique than before the Monte Carlo estimate for $I(\Upsilon)$ is given by

$$I(\Upsilon) = I_0(\Upsilon) + \frac{1}{N}\sum_{j=1}^{N} \frac{\Upsilon(\mathbf{Z}_j) f(\mathbf{Z}_j)}{p(\mathbf{Z}_j)} + \epsilon_{\delta g} \quad \text{with} \quad \epsilon_{\delta g} \simeq \frac{\sigma_{\delta g}}{\sqrt{N}}$$

where

$$I_0(\Upsilon) = \int_V \Upsilon(\mathbf{Z}) f_0(\mathbf{Z}) \, \mathrm{d}\tau$$

can be expressed analytically.

Advantage of δf method

- The function δg is defined as

$$\delta g = \Upsilon(\mathbf{Z})\delta f(\mathbf{Z})/p(\mathbf{Z})$$

while $\sigma_{\delta g}$ the deviation of δg is given by

$$\mathcal{V}(\delta g) = \sigma_{\delta g}^2 = \int_V (\delta g - \mathbb{E}_p(\delta g))^2 p(\mathbf{Z})\, \mathrm{d}\tau \qquad (11)$$

- The advantage of this control variate technique is then evident.
- Indeed, by comparing the error in the Monte Carlo estimates, we see that the noise is reduced by a factor $\delta f/f$ for the same number of sample points.

Limitations of δf method (1/2)

- To conclude, the control variate-δf method reduces noise by reducing the size of the Monte Carlo contribution to $I(\Upsilon)$.
- But it is important to point that this method also concentrates all the relevant physics into this small integral of δf and its time evolution
- The accuracy of the method crucially depends on accurate evaluations of the moments of δf.

Limitations of δf method (2/2)

- For this reason, there are two complementary requirements:
  - (i) Low noise is only accomplished by ensuring that
$$\|f - f_0\|/\|f\| \ll 1$$
  - (ii) Accuracy is only possible if the rel. error in $\delta I(\Upsilon)$ is small, i.e
$$\|\epsilon_{\delta g}/\delta I(\Upsilon)\| \ll 1$$
- The first objective can be realized with a well-chosen control variate f_0 and a small number of macro-particles N
- But the second still requires a large number of markers, since
$$\epsilon_{\delta g}/\delta I(\Upsilon) \sim 1/\sqrt{N}$$

Other major improvements linked to physic

- In addition of these classical Monte Carlo approaches, knowledge of the underlying physics have inspired other major improvements.
- These new techniques of reduction of the noise have been essentially developed in ORB5 code
- Just 2 examples :
  1. An optimized choice of f_0
  2. A field-aligned coordinates filter

An optimized choice of f_0

- It is clear that the choice of $f_0 = f_{eq}$, where f_{eq} represents the initial equilibrium state, is indicated.
- But as f can evolves away from f_{eq}, at least in some regions, use of the latter will no longer provide a small variance estimator.
- An intuitive idea would be to evolve f_0 in such a way to "follow" f.
- One such technique has been implemented in collisional Monte Carlo simulations *[Brunner, PoP (1999)]*.
  - An appropriate choice for $f_0(t)$ was a shifted Maxwellian distribution, evolved using fluid equations
- A more general technique *[Allfrey, CPC (2003)]*.
  - δf directly from the constancy of f along orbits $\mathbf{Z}(t)$ as $\delta f(\mathbf{Z}_j(t)) = f(\mathbf{Z}_j(t_0)) - f_0(\mathbf{Z}_j(t))$.

A field-aligned coordinates filter (1/2)

- Another physics characteristic which has been recently taken into account is the fact that micro-turbulence modes are characterized by very small parallel wave-numbers, $|k_{\|}|\rho_s \leq \rho^*$, due to the gyro-ordering
- Jolliet et al. *[Jolliet, CPC (2007)]* have developed a filter which takes advantage of this strong anisotropy of the perturbations.
- They have shown that filter the modes, which may be present in the simulation but do not satisfy this ordering, is a very efficient way to avoid accumulating significant level of numerical noise.

A field-aligned coordinates filter (1/2)

- This has been performed applying what is called a field-aligned Fourier filter and it is shown that orders of magnitude improvement can be gained.
- The signal to noise ratio depends on the number of markers per Fourier mode retained in the filter and no more on the number of markers per numerical degree of freedom of the field representation