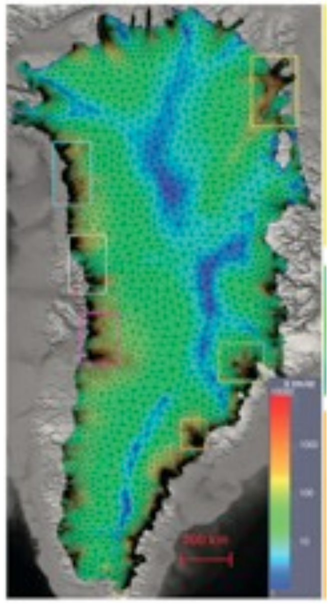
Anisotropic mesh adapation

Before:

pre-processing using **external codes** to :

- **compute metric**
- perform **mesh adaptation** (Yams)



Now:

Elmer Solvers to **:**

- **compute metric**
- compute **metric intersection (multi-variables optimisation)**
- call **remeshing library** (MMG)

Can be used for:

- **pre-processing**
- **transient mesh adaptation** using Elmer internal capabilities for mesh to mesh interpolation

Current limits:

- **2D plane-view** mesh adaptation
- mesh adaptation is **serial**
- **No yet in the Git repository**

Cf poster for description and validation in diagnostic

Anisotropic mesh adaptation for marine ice-sheet modelling

F. Gillet-Chaulet[1], L. Tavard[2], I. Merino[1], V.Peyaud[1], J. Brondex[1], G. Durand[1], and O. Gagliardini[1]

[1]Institut des Géosciences de L'Environnement (IGE), Grenoble, France [2]Infrastructure de Calcul Intensif et de Données (UMS GRICAD), Grenoble, France

Improving forecasts of ice-sheets contribution to sea-level rise requires, amongst others, to **correctly model the dynamics of the grounding line (GL)**. **Grid refinement** in the GL vicinity is a key component to obtain reliable results. **Improving model accuracy** while maintaining **the computational cost affordable** has then been an important target for the development of marine ice-sheet models. **Adaptive mesh refinement (AMR)** is a method where the accuracy of the solution is controlled by spatially adapting the mesh size. The main difficulty is to find **efficient and reliable estimators** of the numerical error to control the mesh size. Here, we use an error estimate based on the **interpolation error**, proposed by Frey and Alauzet (2005). Routines to compute the anisotropic metric defining the mesh size have been implemented in the finite element model **Elmer/Ice** (Gagliardini et al., 2013). The mesh adaptation is performed using the freely available library **MMG** (Dapogny et al., 2014) called from Elmer/Ice. Our implementation in restricted to the adaptation of **plane-view 2D meshes** comprised of **linear 3-nodes triangular elements.**

We show that **combining** the information from **the ice thickness**, **the ice velocity** and **the basal drag** to define the mesh size, allows to **reduce the number of mesh nodes** by more than **one order of magnitude**, for **the same numerical accuracy**, when compared to uniform mesh refinement. For transient simulations, we have implemented an iterative transient mesh adaptation algorithm (Alauzet et al., 2007) that allows to refine the mesh in the area where the GL evolves.

1. Anisotropic metric definition

The metric M, used to define the element size, derives from a **geometric error estimate** based on an **upper bound** for the **interpolation error** of a continuous field to piecewise linear elements. For a variable v, M depends on the eigenvalues λ_i and eigenvector matrix R of the hessian matrix of v, H:

$$M = R.\tilde{\Lambda}.R^{-1}, \text{ with } \tilde{\Lambda} = \begin{pmatrix} \tilde{\lambda}_1 & 0 \\ 0 & \tilde{\lambda}_2 \end{pmatrix}$$

where

$$\tilde{\lambda}_i = \min\left(\max\left(\frac{c|\lambda_i|}{\epsilon_v}, \frac{1}{l_{\max}^2}\right), \frac{1}{l_{\min}^2}\right)$$

with:

- c a geometric constant equal to 2/9 in 2D
- l_{min} (resp. l_{max}) is a prescribed minimal (resp. maximal) edge size
- ϵ_v **the prescribed maximum error**

2. Metric intersection

Sometimes it can be desirable to use several variables to compute the metric, if we want, for example, to capture with the same mesh different physical phenomena represented by different variables

The intersection $M_{1\cap 2} = M_1 \cap M_2$ of two metrics M_1 and M_2 is given by (Alauzet *et al.*, 2007):

$$M_{1\cap 2} = {}^T P^{-1} \begin{pmatrix} \max(\mu_1^1, \mu_1^2) & 0 \\ 0 & \max(\mu_2^1, \mu_2^2) \end{pmatrix} P^{-1}$$

with P the matrix where the columns are the normalised eigenvectors $\{e_i\}_{i=1,2}$, of $N = M_1^{-1}M_2$ and $\mu_i^j = {}^T e_i M_j e_i$

Geometrically speaking, this reduction gives the largest ellipsoid included in the geometrical intersection of the two ellipsoids associated to M_1 and M_2

3. Experimental setup

To illustrate the performance of the mesh adaptation we use the depth-averaged **SSA equations**. We use the experimental set-up proposed by the intercomparison exercise **MISMIP+** (Asay-Davis et al., 2016).

We construct a **reference simulation** by running the model with constant forcing and boundary conditions until a quasi steady-state is reached.

We use a structured mesh made of linear triangular elements where the nodes are regularly spaced every 250 m along x and y, resulting in N_{Ref} = **822081 nodes**.

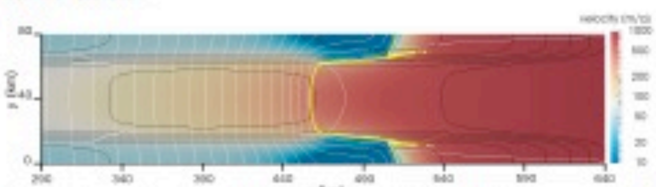


Fig. 1: Reference velocity used for the mesh adaptation; black lines are bedrock elevation contours every 100m; white lines are surface elevation contours every 50m; yellow line is the grounding line

4. Diagnostic Experiment

In this experiment, the mesh is adapted using the reference simulation (Fig.1) as perfect observations. The ice sheet topography is interpolated on the adapted mesh and the flow velocities computed. ***We have 2 objectives:***

- verify that ϵ_v allows to effectively control the numerical error, *i.e.* the solution on the adapted mesh converges to the reference solution when ϵ_v is decreased.
- find the variable or combination of variables that leads to the smallest number of mesh nodes for a given accuracy.

To measure the accuracy of the solution on the adapted mesh, mean **thickness and velocity errors** are defined as:

$$\bar{\epsilon}_h = \sqrt{\frac{1}{N_{ref}} \sum_{i=1}^{N_{ref}} (h - h_{ref})^2} \qquad \bar{\epsilon}_u = \sqrt{\frac{1}{N_{ref}} \sum_{i=1}^{N_{ref}} |u - u_{ref}|^2}$$

where h and u are the thickness and velocity magnitude linearly interpolated from the adapted mesh to the reference mesh.

- ***Validation:***
  - As expected, **the error on** h, is only **an interpolation error** and **is effectively controlled by the mesh adaptation scheme.**
  - **The error on the velocity** results from **discretisation errors of the ice-sheet topography** and **boundary conditions**. It effectively decreases with the prescribed error ϵ_v but it is one order of magnitude higher.

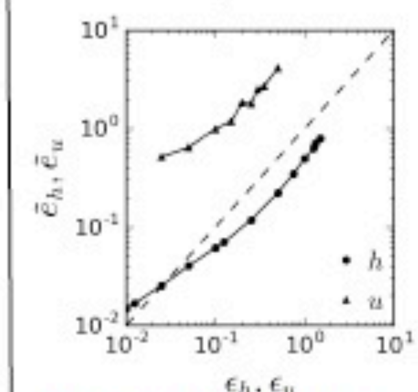


Fig. 2 Thickness and velocity errors as a function of the prescribed error when the mesh is adapted using, respectively, h and u.

- ***One variable adaptation:***

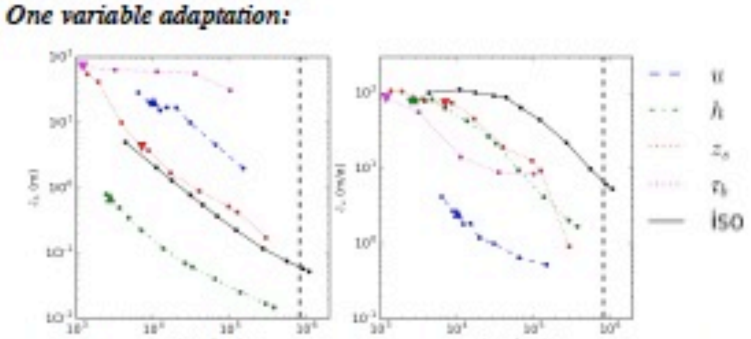


Fig. 3: Thickness and velocity errors as a function of the number of nodes when the mesh is adapted using the reference solution for u, h, z_s, τ_b and unstructured mesh with prescribed uniform element size

- ***Multi-variable adaptation:***

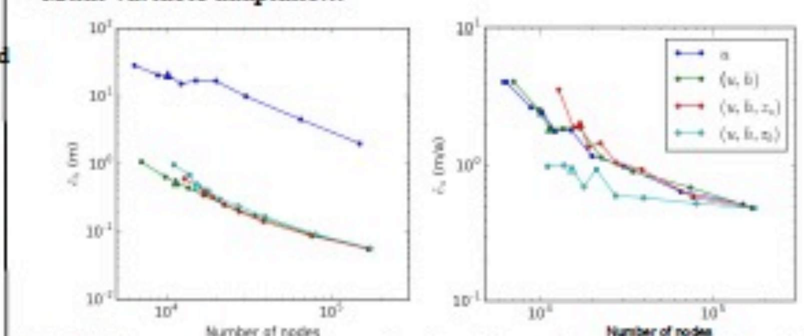


Fig. 4: Thickness and velocity errors as a function of the number of nodes when the mesh is adapted using the reference solution different combinations of variables.

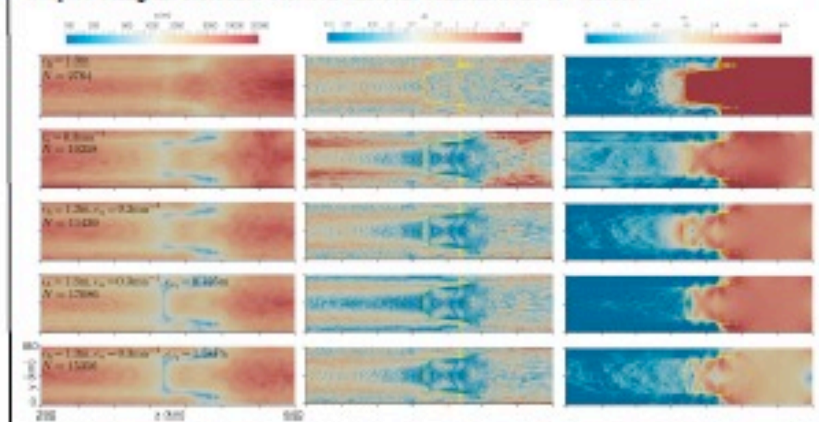


Fig. 5 : (columns from left to right) Mean element size, local error on h and u when the mesh is adapted using the reference solution for (rows from top to bottom) (h), (u), (u,h), (u,h,z_s), (u,h,τ_b). ϵ_v for $v = h, u, z_s, \tau_b$ are the values of the interpolation error prescribed for the mesh adaptation. N is the number of mesh nodes. Results shown here are marked with symbol (▲) in Fig. 3 and Fig. 4

5. Transient Experiment

Starting from the steady state, the friction coefficient is divided by a factor 2 and the model is run for 25 years. The GL moves and eventually **exits the refined area** (Fig. 6, left). We have implemented a **transient mesh adaptation algorithm** (Alauzet et al., 2007) where the transient simulation is reiterated, allowing to **anticipate the GL displacement** and **to adapt the mesh to the transient solution** using the metric intersection formulae (Fig. 6, right)

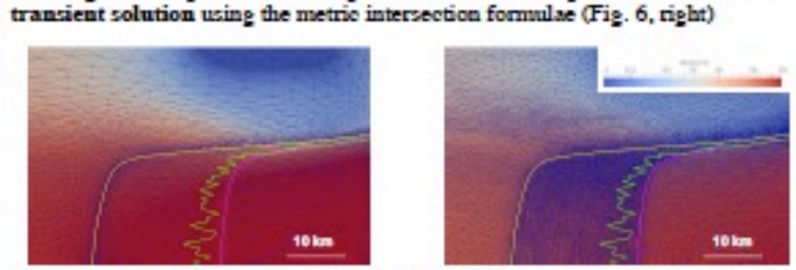
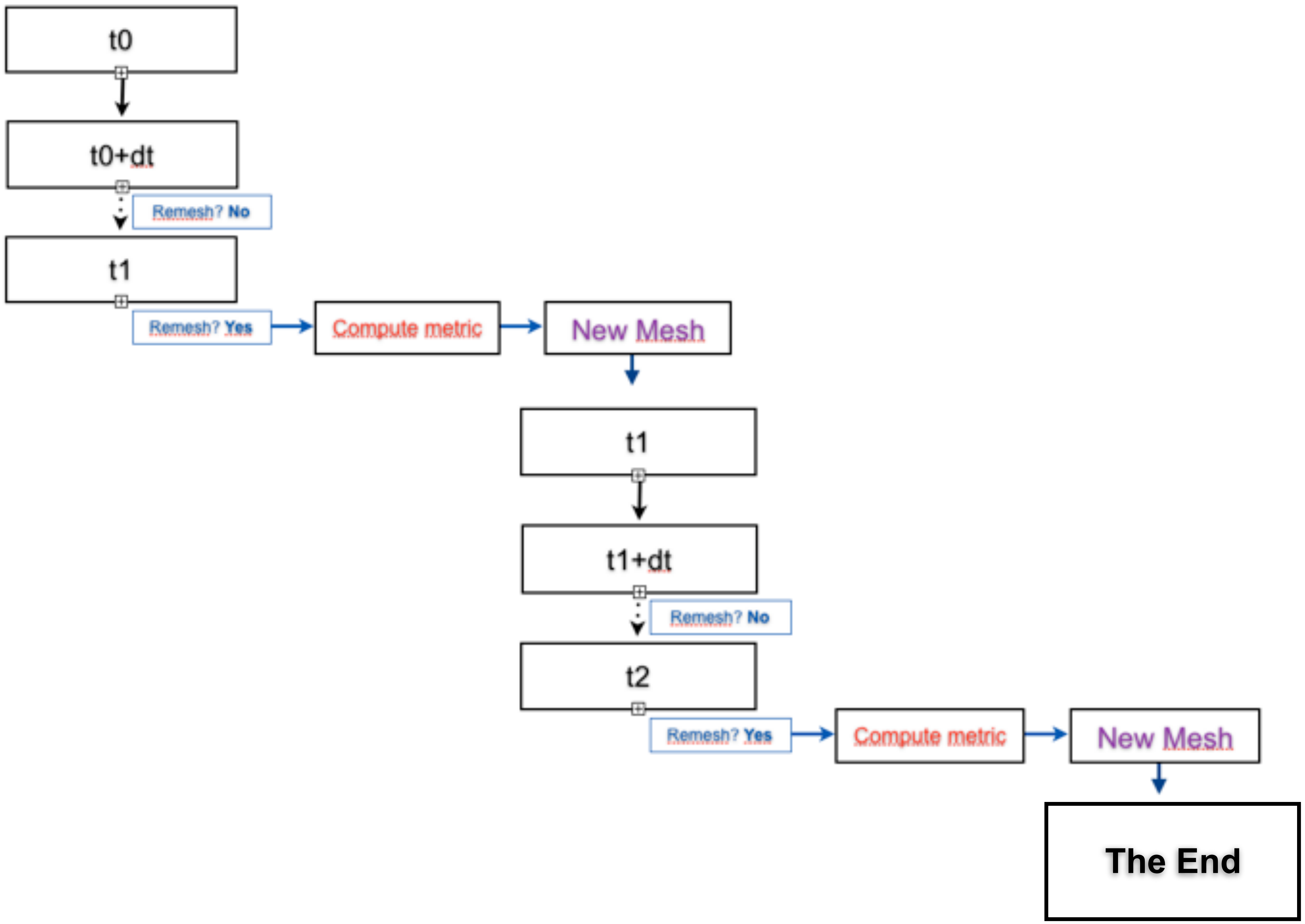


Fig. 6: (Left) mesh adapted at t=0 and GL at t=25a (green); (Right) mesh adapted on the transient run (5 adaptation iterations) and GL at t=25a (magenta)

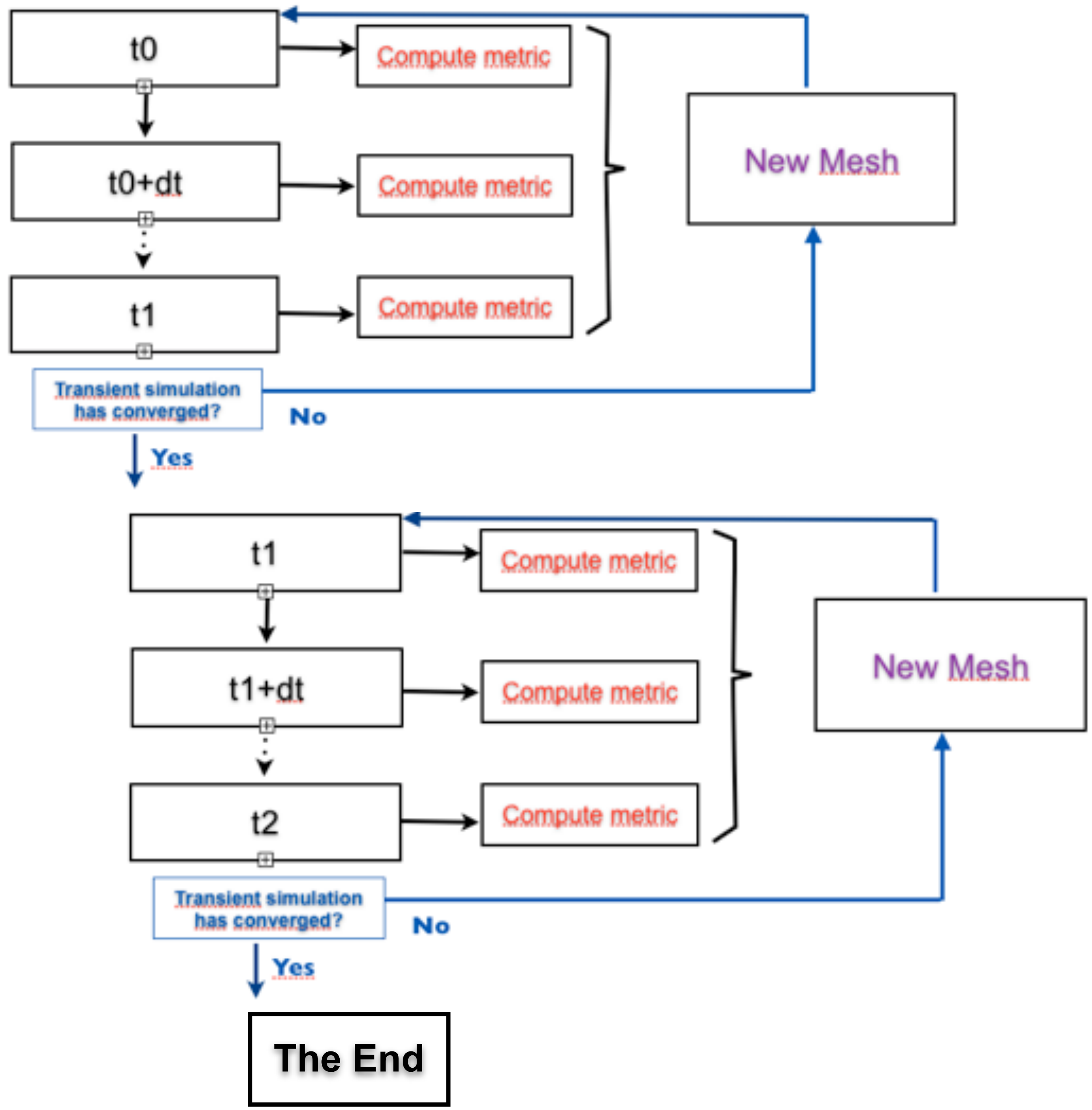
Transient mesh adapation



limits:

- **Remeshing criteria** (every n timesteps, GL maximum displacement, ...) ?
- Mesh is adapted using diagnostic informations

Fixed point transient algoritm



Need to **iterate** the transient simulation -> additionnal computationnal cost
but:

- Mesh is adapated using **transient informations** (metric intersection formulae)
- **Convergence test** is **intrisic** to the procedure