

# Optimisation de trajectoire : optimisation de la largeur entre les droites avec contrainte de température

Mathilde Boissier

Avril 2018

## Modèle et objectif

On s'intéresse à la construction d'une pièce rectangulaire  $\Omega$  par méthode SLM ( $\partial\Omega = \Gamma_N \cup \Gamma_D$ ,  $\Gamma_N \cap \Gamma_D$ ). On souhaite donc faire passer une source de chaleur le long d'une trajectoire prédefinie afin de faire fusionner le lit de poudre. Cependant, il ne faut pas que la température puisse être trop élevée sous peine de créer des contraintes résiduelles thermiques qui nuiraient à la qualité de la pièce.

Le travail présenté dans ce document consiste à, ayant choisi une trajectoire de type ligne droite pour le laser, optimiser la distance entre les lignes afin de fusionner la poudre ( $\forall x \in \Omega$ ,  $\exists t \in [0, t_F]$  tel que  $T(x, t) > T_{phase}$ ) tout en maîtrisant la température dans le solide ( $\forall x \in \Omega$ ,  $\forall t \in [0, t_F]$ ,  $T(x, t) < T_{sup}$ ).

## Modélisation thermique

On commence par introduire le modèle thermique utilisé ici. Il est semblable à celui utilisé dans l'article de Lukas. On rajoute ici de la diffusion car notre modèle est seulement 2D et on veut tout de même modéliser la perte de chaleur dans les couches inférieures. Les conditions aux limites sont les conditions de Dirichlet car autour du solide, on a uniquement de la poudre qui est adiabatique. On cherche un champ de température  $T(x, t) : [0, t_F] \times D \rightarrow \mathbb{R}$  tel que :

$$\begin{cases} \rho \partial_t T + \frac{\lambda}{ep_{car}^2} (T - T_{ini}) - \operatorname{div}(\lambda \nabla T) = Q & \text{in } (0, t_F) \times D \\ T = T_{ini} & \text{in } (0, t_F) \times \partial\Omega \\ T(0) = T_{ini} & \text{in } \Omega \end{cases} \quad (1)$$

Le solide et la poudre ont des valeurs différentes pour les paramètres physiques suivants (en notant  $\chi_{solid}$  la fonction caractéristique du solide) :

$$\begin{aligned} \rho &= \rho_{solid} * \chi_{solid} + \rho_{powder} * (1 - \chi_{solid}) \\ \lambda &= \lambda_{solid} * \chi_{solid} + \lambda_{powder} * (1 - \chi_{solid}) \\ \beta &= \beta_{solid} * \chi_{solid} + \beta_{powder} * (1 - \chi_{solid}) \end{aligned} \quad (2)$$

## Trajectoire de la source

On considère une trajectoire en ligne droite comme sur le schéma ci-dessous (Figure ??). On estime que la vitesse du laser est constante et égale à 1. On prend pour pas de temps  $\delta t = 2 * \delta x$ . La source est modélisée par une gaussienne centrée en un point (Figure ??).

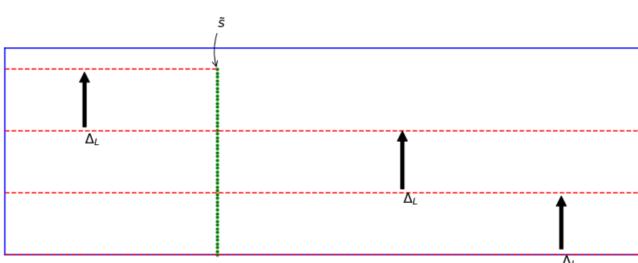


Figure 1: Trajectoire de la source

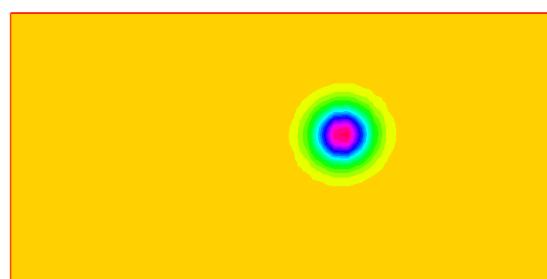


Figure 2: forme de la source

On modélise dans un premier temps le temps pour changer de ligne comme nul. De plus, on considère un temps final  $t_F$ , temps nécessaire pour effectuer toute la trajectoire comme constant. On estime que, que les lignes droites

soient "dans le domaine de calcul ou non", la source va y passer. Ainsi, on décorrèle le temps final de la trajectoire de lasage ce qui permet de le considérer comme un paramètre fixe lors de l'optimisation (Figure XXX).

Afin de modéliser l'arrêt de la source lorsqu'on sort de la zone de travail, on choisit de multiplier la source à une fonction de heaviside qui s'annule lorsque le laser sort de la zone de travail. Ainsi, la zone continue de parcourir "pour de faux" des lignes hors du domaine de travail sans perturber la thermique. Cette approche induit deux discontinuités différentes :

- discontinuité de la source avec la fonction de heaviside.
- discontinuité des paramètres de la fonction de heaviside : le temps  $t_s$  auquel la source sort du domaine de travail et s'éteint est fortement discontinu par rapport à l'écartement entre les lignes  $\Delta L$  (il s'agit même d'une fonction en escaliers).

On peut éliminer la deuxième discontinuité en introduisant une deuxième variable,  $\tilde{s}$  à optimiser et qui représente la longueur parcourue par le laser. Cette longueur n'est pas nécessairement proportionnelle à la longueur d'une ligne et donc on peut estimer que la source doit s'éteindre au milieu de la dernière ligne. La source est alors modélisée par une gaussienne centrée en un point mouvant, dépendant du temps et de l'écartement entre les lignes  $\Delta L$  et d'une fonction de heaviside qui vaut 0 lorsque l'on a "dépassé l'abscisse curviligne finale  $\tilde{s}$ " :  $1_{tV < \tilde{s}}$ . On a toujours ici la discontinuité de la source avec la fonction de heaviside. Cependant, l'abscisse curviligne est, elle, continue, ce qui nous évite la deuxième discontinuité.

## Theorie

### Probleme

On veut minimiser les endroits n'ayant pas subi de changement phase tout en ne dépassant jamais une valeur seuil de la température ( $T_{sup}$ ), en tout point et en tout instant de fabrication.

Les paramètre d'optimisation sont :

- la distance entre deux lignes de parcours du laser  $\Delta L$ . On borne  $\Delta L$  par 0 et par la largeur de la pièce ( $\Delta L \in [0, L_{max}]$ ).
- l'"abscisse curviligne" finale  $\tilde{s}$  qui détermine l'arrêt de la source. On borne  $\tilde{s}$  par 0 et par un nombre de ligne maximal (déterminé par le temps final constant  $t_F$  choisi) multiplié par le temps de parcours d'une seule ligne.

### Equation de la chaleur

On pose  $\tilde{T} = T - T_{ini}$  (i.e.  $T = \tilde{T} + T_{ini}$ ). On a alors, d'après l'équation 1 (l'arrêt de la source est modélisée par la fonction source  $Q$  elle-même. On écrit donc bien l'équation jusqu'à  $t_F$  fixé) :

$$\begin{cases} \left( \rho \partial_t \tilde{T} + \frac{\lambda}{ep_{car}^2} \tilde{T} - \operatorname{div}(\lambda \nabla \tilde{T}) \right) = Q(\Delta L, \tilde{s}) & \text{in } (0, t_F) \times D \\ \tilde{T} = 0 & \text{in } (0, t_F) \times \Gamma_D \\ \tilde{T}(0) = 0 & \text{in } \Omega \end{cases} \quad (3)$$

Le problème variationnel lié à la chaleur est alors le suivant :

trouver  $\tilde{T} \in V = \{w \in H^1([0, t_M], \Omega) \text{ tel que } w|_{\Gamma_D} = 0\}$  tel que

$$\forall v \in V, \forall t \in [0, t_M], \quad \int_{\Omega} \rho \partial_t \tilde{T} v + \frac{\lambda}{ep_{car}^2} \tilde{T} v dx + \int_{\Omega} \lambda \nabla \tilde{T} \nabla v dx - \int_{\Omega} Q(\Delta L, \tilde{s}) v dx = 0 \quad (4)$$

### Changement de phase

On souhaite ici que :

$$\forall x \in \Omega, \quad \left[ (\|T\|_{\infty} - T_{fu})^{-} \right]^2 = 0 \quad (5)$$

On approxime la norme infinie par une norme  $r$  et on a donc la fonction objectif suivante :

$$J(\Delta L, \tilde{s}) = \int_{\Omega} \left[ \left( \left( \int_0^{t_F} |T|^r dt \right)^{\frac{1}{r}} - T_{fu} \right)^{-} \right]^2 = \int_{\Omega} \left[ \left( \left( \int_0^{t_M} |\tilde{T} + T_{ini}|^r dt \right)^{\frac{1}{r}} - T_{fu} \right)^{-} \right]^2 \quad (6)$$

## Contrainte

On veut éviter que la température dépasse une température seuil  $T_{sup}$ . On impose pour cela une contrainte :

$$C(\Delta L, \tilde{s}) = \int_0^{t_F} \int_{\Omega} [(T - T_{sup})^+]^2 dx dt - tol_{sup} = \int_0^{t_M} \int_{\Omega} [(\tilde{T} + T_{ini} - T_{sup})^+]^2 dx dt - tol_{sup} \quad (7)$$

## Source

La source est une gaussienne qui parcourt la trajectoire et qui s'éteint quand l'"abscisse" dépasse l'"abscisse curviline" finale  $\tilde{s}$ . On note  $t$  le temps ( $t = \Delta t * it$ ),  $t_{Li}$  le temps pour parcourir une ligne,  $V$  la vitesse de parcours,  $(x_C(t), y_C(t, \Delta L))$  le "centre de la gaussienne" à l'instant  $t$ . On a alors :

$$\begin{aligned} x_C(t) &= \left( t - \text{Ent}\left[\frac{t}{t_{Li}}\right] * t_{Li} \right) \\ Y_C(T) &= \text{Ent}\left[\frac{t}{t_{Li}}\right] * \Delta L \end{aligned} \quad (8)$$

On pose alors la fonction  $S(t, x, y, \Delta L) = P \exp \left( -100 \left( (x - x_C(t))^2 + (y - y_C(t, \Delta L))^2 \right) \right)$

On modélise l'action d'éteindre la source par la fonction de heaviside régularisée suivante :

$$h_\epsilon(t) = \begin{cases} 0 & \text{si } t < -\epsilon \\ 0.5 \left( 1 + \frac{t}{\epsilon} + \frac{1}{\pi} \sin(\pi \frac{t}{\epsilon}) \right) & \text{si } -\epsilon < t < \epsilon \\ 1 & \text{si } \epsilon < t \end{cases}$$

Cette fonction est  $\mathcal{C}^1$ , avec

$$h'_\epsilon(t) = \begin{cases} 0 & \text{si } t < -\epsilon \\ \frac{1}{2\epsilon} \left( 1 + \cos(\pi \frac{t}{\epsilon}) \right) & \text{si } -\epsilon < t < \epsilon \\ 0 & \text{si } \epsilon < t \end{cases}$$

On a alors

$$Q(t, x, y, \Delta L, \tilde{s}) = S(t, x, y, \Delta L) * h_\epsilon(\tilde{s} - tV) \quad (9)$$

## Problème d'optimisation

On s'intéresse donc au problème suivant :

$$\min_{\substack{\Delta L \in [0, L_{max}] \\ \tilde{s} \in [0, S_{max}]}} J(\Delta L, \tilde{s}) = \int_{\Omega} \left[ \left( \left( \int_0^{t_F} |\tilde{T} + T_{ini}|^r dt \right)^{\frac{1}{r}} - T_{fu} \right)^{-} \right]^2$$

$$\text{tel que } C(\Delta L, \tilde{s}) = \int_0^{t_F} \int_{\Omega} [(\tilde{T} + T_{ini} - T_{sup})^+]^2 dx dt - tol_{sup} \leq 0$$

avec  $\tilde{T}$  solution de l'Equation 3.

## Algorithme du Lagrangien

On va tout d'abord tester l'algorithme d'Uzawa pour gérer la contrainte. On pose  $\text{lag}_{sup}$  le multiplicateur de Lagrange. On s'intéresse au problème modifié :

$$\min_{\substack{\Delta L \in [0, L_{max}] \\ \tilde{s} \in [0, S_{max}]}} J(\Delta L, \tilde{s}) \quad (10)$$

avec

$$\tilde{J}(\Delta L, \tilde{s}) = \int_{\Omega} \left[ \left( \left( \int_0^{t_F} |\tilde{T} + T_{ini}|^r dt \right)^{\frac{1}{r}} - T_{fu} \right)^{-} \right]^2 + \text{lag}_{sup} \left[ \int_0^{t_F} \int_{\Omega} [(\tilde{T} + T_{ini} - T_{sup})^+]^2 dx dt - tol_{sup} \right] \quad (11)$$

et on aura à chaque itération

$$\text{lag}_{sup}^{n+1} = \begin{cases} 0 & \text{si } C(\Delta L^n) \leq 0 \\ \text{lag}_{sup}^n + \mu_{lag} C(\Delta L^n, \tilde{s}^n) & \text{si } C(\Delta L^n, \tilde{s}^n) > 0 \end{cases} \quad (12)$$

### Augmented Lagrangian Method (ALM)

Le deuxième algorithme testé est celui du Lagrangien augmenté (ALM dans la suite). (VOIR NOCEDAL XXX)

On pose

$$\psi(c, \lambda; \mu) = \begin{cases} -\lambda * c + \frac{1}{2\mu} * c^2 & \text{si } c - \lambda * \mu \leq 0 \\ -\frac{\mu}{2} \lambda^2 & \text{sinon} \end{cases} \quad (13)$$

On a alors, avec  $\lambda$  paramètre à mettre à jour et  $\mu$  paramètre de pénalisation fixé, une nouvelle fonction objectif du type :

$$\tilde{J}(\Delta L, \tilde{s}) = \int_{\Omega} \left[ \left( \left( \int_0^{t_F} |\tilde{T} + T_{ini}|^r dt \right)^{\frac{1}{r}} - T_{fu} \right)^{-} \right]^2 + \psi \left( \text{tol}_{sup} - \left[ \int_0^{t_F} \int_{\Omega} [(\tilde{T} + T_{ini} - T_{sup})^+]^2 dx dt \right], \lambda; \mu \right) \quad (14)$$

### Calcul des dérivées : méthode sans adjoint pour la température

On veut trouver la dérivée de  $\tilde{J}$  par rapport à  $\Delta L$  et  $\tilde{s}$ . On procède alors de la manière suivante :

#### Cas de $\Delta L$

$\tilde{J}(\Delta L, \tilde{s}) = \tilde{j}(\tilde{T}(\Delta L, \tilde{s}))$ . On a donc :

$$\partial_{\Delta L} \tilde{J}(\Delta L, \tilde{s}) = \partial_{\tilde{T}} \tilde{j}(\tilde{T}(\Delta L, \tilde{s})) (\partial_{\Delta L} \tilde{T}(\Delta L, \tilde{s})) \quad (15)$$

En dérivant l'équation vérifiée par  $\tilde{T}$  en fonction de  $\Delta L$ , on obtient l'équation vérifiée par  $\partial_{\Delta L} \tilde{T}$  :

$$\begin{cases} \left( \rho \partial_t \partial_{\Delta L} \tilde{T} + \frac{\lambda}{ep_{car}^2} \partial_{\Delta L} \tilde{T} - \text{div}(\lambda \nabla \partial_{\Delta L} \tilde{T}) \right) = \partial_{\Delta L} Q(\Delta L, \tilde{s}) & \text{in } (0, t_F) \times D \\ \partial_{\Delta L} \tilde{T} = 0 & \text{in } (0, t_F) \times \Gamma_D \\ \partial_{\Delta L} \tilde{T}(0) = 0 & \text{in } \Omega \end{cases} \quad (16)$$

et le problème variationnel associé ainsi que sa discréétisation en temps sont similaires à ceux liés à  $\tilde{T}$ .

On a aussi accès à la dérivée de la fonction objectif de changement de phase par rapport à  $\tilde{T}$  :

$$\partial_{\tilde{T}} J(\tilde{T})(\phi) = \int_{\Omega} 2 \left( \left( \int_0^{t_F} |\tilde{T} + T_{ini}|^r dt \right)^{\frac{1}{r}} - T_{fu} \right)^{-} \frac{1}{r} \left( \int_0^{t_F} |\tilde{T} + T_{ini}|^r dt \right)^{\frac{1}{r}-1} \int_0^{t_F} r |\tilde{T} + T_{ini}|^{r-1} \phi dt dx \quad (17)$$

De même, on a accès à la dérivée de la contrainte par rapport à  $\tilde{T}$  :

$$\partial_{\tilde{T}} C(\tilde{T})(\phi) = \int_0^{t_F} \int_{\Omega} 2[(\tilde{T} + T_{ini} - T_{sup})^+] \phi dx dt \quad (18)$$

- Algorithme d'Uzawa : On obtient alors la dérivée de la fonction objectif par rapport à  $\tilde{T}$ :

$$\partial_{\tilde{T}} \tilde{j}(\tilde{T})(\phi) = \partial_{\tilde{T}} J(\Delta L, \tilde{s})(\phi) + \text{lag}_{sup} \partial_{\tilde{T}} C(\Delta L, \tilde{s})(\phi) \quad (19)$$

et donc la dérivée pour  $\tilde{J}$  par rapport à  $\Delta L$  :

$$\partial_{\Delta L} \tilde{J} = \partial_{\tilde{T}} J(\Delta L, \tilde{s})(\partial_{\Delta L} \tilde{T}) + \text{lag}_{sup} \partial_{\tilde{T}} C(\Delta L, \tilde{s})(\partial_{\Delta L} \tilde{T}) \quad (20)$$

- ALM : On obtient alors la dérivée de la fonction objectif par rapport à  $\tilde{T}$ :

$$\partial_{\tilde{T}} \tilde{j}(\tilde{T})(\phi) = \partial_{\tilde{T}} J(\Delta L, \tilde{s})(\phi) + \partial_1 \psi(\text{tol}_{sup} - C(\Delta L, \tilde{s}), \lambda; \mu) \partial_{\tilde{T}} C(\Delta L, \tilde{s})(\phi) \quad (21)$$

et donc la dérivée pour  $\tilde{J}$  par rapport à  $\Delta L$  :

$$\partial_{\Delta L} \tilde{J} = \partial_{\tilde{T}} J(\Delta L, \tilde{s})(\partial_{\Delta L} \tilde{T}) + \partial_1 \psi(\text{tol}_{sup} - C(\Delta L, \tilde{s}), \lambda; \mu) \partial_{\tilde{T}} C(\Delta L, \tilde{s})(\partial_{\Delta L} \tilde{T}) \quad (22)$$

## Cas de $\tilde{s}$

$\tilde{J}(\Delta L, \tilde{s}) = \tilde{j}(\tilde{T}(\Delta L, \tilde{s}))$ . On a donc :

$$\partial_{\tilde{s}} \tilde{J}(\Delta L, \tilde{s}) = \partial_{\tilde{T}} \tilde{j}(\tilde{T}(\Delta L, \tilde{s})) (\partial_{\tilde{s}} \tilde{T}(\Delta L, \tilde{s})) \quad (23)$$

En dérivant l'équation vérifiée par  $\tilde{T}$  en fonction de  $\Delta L$ , on obtient l'équation vérifiée par  $\partial_{\tilde{s}} \tilde{T}$  :

$$\begin{cases} \left( \rho \partial_t \partial_{\tilde{s}} \tilde{T} + \frac{\lambda}{ep_{car}^2} \partial_{\tilde{s}} \tilde{T} - \operatorname{div}(\lambda \nabla \partial_{\tilde{s}} \tilde{T}) \right) = \partial_{\tilde{s}} Q(\Delta L, \tilde{s}) & \text{in } (0, t_F) \times D \\ \partial_{\tilde{s}} \tilde{T} = 0 & \text{in } (0, t_F) \times \Gamma_D \\ \partial_{\tilde{s}} \tilde{T}(0) = 0 & \text{in } \Omega \end{cases} \quad (24)$$

et le problème variationnel associé ainsi que sa discrétisation en temps sont similaires à ceux liés à  $\tilde{T}$ .

On a les mêmes dérivées par rapport à  $\tilde{T}$  pour la fonction de changement de phase et la contrainte que dans le cas de  $\Delta L$ .

- Algorithme d'Uzawa : On obtient alors la dérivée de la fonction objectif par rapport à  $\tilde{T}$ :

$$\partial_{\tilde{T}} \tilde{j}(\tilde{T})(\phi) = \partial_{\tilde{T}} J(\Delta L, \tilde{s})(\phi) + \operatorname{lag}_{sup} \partial_{\tilde{T}} C(\Delta L, \tilde{s})(\phi) \quad (25)$$

et donc la dérivée pour  $\tilde{J}$  par rapport à  $\tilde{s}$  :

$$\partial_{\tilde{s}} \tilde{J} = \partial_{\tilde{T}} J(\Delta L, \tilde{s})(\partial_{\tilde{s}} \tilde{T}) + \operatorname{lag}_{sup} \partial_{\tilde{T}} C(\Delta L, \tilde{s})(\partial_{\tilde{s}} \tilde{T}) \quad (26)$$

- ALM : On obtient alors la dérivée de la fonction objectif par rapport à  $\tilde{T}$ :

$$\partial_{\tilde{T}} \tilde{j}(\tilde{T})(\phi) = \partial_{\tilde{T}} J(\Delta L, \tilde{s})(\phi) + \partial_1 \psi(\operatorname{tol}_{sup} - C(\Delta L, \tilde{s}), \lambda; \mu) \partial_{\tilde{T}} C(\Delta L, \tilde{s})(\phi) \quad (27)$$

et donc la dérivée pour  $\tilde{J}$  par rapport à  $\tilde{s}$  :

$$\partial_{\tilde{s}} \tilde{J} = \partial_{\tilde{T}} J(\Delta L, \tilde{s})(\partial_{\tilde{s}} \tilde{T}) + \partial_1 \psi(\operatorname{tol}_{sup} - C(\Delta L, \tilde{s}), \lambda; \mu) \partial_{\tilde{T}} C(\Delta L, \tilde{s})(\partial_{\tilde{s}} \tilde{T}) \quad (28)$$

## Discrétisation temporelle

On utilise un schéma implicite pour l'équation de la chaleur :

$$\begin{cases} \int_{\Omega} \left( \rho + \frac{\lambda}{ep_{car}^2} \Delta t \right) \tilde{T}_{j+1} v dx + \int_{\Omega} \lambda \Delta t \nabla \tilde{T}_{j+1} \nabla v dx - \int_{\Omega} \left( Q_{j+1} \Delta t - \rho \tilde{T}_j \right) v dx = 0 \\ \tilde{T}_0 = T_{ini} \end{cases} \quad (29)$$

$$J(\Delta L, \tilde{s}) = \int_{\Omega} \left[ \left( \left( \sum_{j=0}^{N_F} \Delta t |\tilde{T}_j + T_{ini}|^r \right)^{\frac{1}{r}} - T_{fu} \right)^2 \right] \quad (30)$$

$$C(\Delta L, \tilde{s}) = \sum_{j=0}^{N_F} \Delta t \int_{\Omega} [(\tilde{T}_j + T_{ini} - T_{sup})^+]^2 dx - \operatorname{tol}_{sup} \quad (31)$$

$$\partial_{\Delta L} J(\Delta L, \tilde{s}) = \int_{\Omega} 2 \left( \left( \sum_{j=0}^{N_F} \left( \Delta t |\tilde{T}_j + T_{ini}|^r \right)^{\frac{1}{r}} - T_{fu} \right)^{-} \frac{1}{r} \left( \sum_{j=0}^{N_F} \left( \Delta t |\tilde{T}_j + T_{ini}|^r \right)^{\frac{1}{r}-1} \sum_{j=0}^{N_F} \Delta t \left( r |\tilde{T}_j + T_{ini}|^{r-1} \partial_{\Delta L} \tilde{T} \right) dx \right) \quad (32)$$

$$\partial_{\tilde{s}} C(\Delta L, \tilde{s}) = \sum_{j=0}^{N_F} \Delta t \left( \int_{\Omega} 2 [(\tilde{T}_j + T_{ini} - T_{sup})^+] \partial_{\tilde{s}} \tilde{T} \right) dx \quad (33)$$

## Algorithme

On pose

$$\begin{cases} Norme^r = \int_0^{t_F} |\tilde{T} + T_{ini}|^r dt = \sum_{i=0}^{N_{t_F}} \Delta t |\tilde{T}_i + T_{ini}|^r \\ NormeMoins = \min(Norme - T_{fu}, 0) \end{cases} \quad (34)$$

Lorsqu'on simule le passage du laser, à chaque itération, on calcule :

- la source  $Q$  et la température  $\tilde{T}$
- les dérivées par rapport à  $\Delta L$  et  $\tilde{s}$  de la source et les dérivées de la température  $\partial_{\Delta L}\tilde{T}$  et  $\partial_{\tilde{s}}\tilde{T}$
- $Norme = Norme + \Delta t |\tilde{T}_i + T_{ini}|^r$
- $derPhaseDeltaL = derPhaseDeltaL + \Delta t |\tilde{T}_i + T_{ini}|^{r-1} * \partial_{\Delta L}\tilde{T}$
- $derPhaseStilde = derPhaseStilde + \Delta t |\tilde{T}_i + T_{ini}|^{r-1} * \partial_{\tilde{s}}\tilde{T}$
- $Contrainte = Contrainte + \Delta t \int_{\Omega} \left( (\tilde{T} + T_{ini} - T_{sup})^+ \right)^2 dx$
- $derSupDeltaL = derSupDeltaL + \Delta t \int_{\Omega} 2 \left( \tilde{T} + T_{ini} - T_{sup} \right)^+ \partial_{\Delta L}\tilde{T}$
- $derSupStilde = derSupStilde + \Delta t \int_{\Omega} 2 \left( \tilde{T} + T_{ini} - T_{sup} \right)^+ \partial_{\tilde{s}}\tilde{T}$

A la fin de cette simulation, on calcule :

- $Norme = (Norme)^{1/r}$
- $NormeMoins = \min(Norme - T_{fu}, 0)$
- $J = \int_{\Omega} (NormeMoins^2 dx)$
- $derPhaseDeltaL = \int_{\Omega} 2 * NormeMoins * Norme^{1-r} * derPhaseDeltaL dx$
- $derPhaseStilde = \int_{\Omega} 2 * NormeMoins * Norme^{1-r} * derPhaseStilde dx$
- Dans le cas de Uzawa :
  - $L = J + Lag * Contrainte$
  - $\partial_{\Delta L}L = derPhaseDeltaL + Lag * derSupDeltaL$
  - $\partial_{\tilde{s}}L = derPhaseStilde + Lag * derSupStilde$
- Dans le cas de ALM :
  - $L = J + \psi(\text{tol}_{sup} - Contrainte, Lag, \mu)$
  - $\partial_{\Delta L}L = derPhaseDeltaL + \partial_1 \psi(\text{tol}_{sup} - Contrainte, Lag, \mu) derSupDeltaL$
  - $\partial_{\tilde{s}}L = derPhaseStilde + \partial_1 \psi(\text{tol}_{sup} - Contrainte, Lag, \mu) derSupStilde$

## Pas de descente

On consièrre un coefficient pour chaque variable ( $coef_{\Delta L}$ ,  $coef_{\tilde{s}}$ ). Il est initialisé à l'écart maximum que peuvent prendre les valeurs divisé par 10 :

$$coef_{DeltaL} = (\Delta L_{max} - \Delta L_{min}) / 10 \quad (35)$$

. A chaque itération, le pas de descente est calculé de manière à ce que la différence entre l'ancienne valeur et la nouvelle soit de la valeur du coefficient. On aura donc :

$$|\Delta L_{i+1} - \Delta L_i| = coef_{\Delta L} \quad (36)$$

Ce coefficient est divisé par 5 lorsque l'itération est ratée et multiplié par 1.5 lorsque l'itération est réussie.

# Tests et résultats

## Algorithme d'Uzawa

Phase seule

Contrainte seule

Objectif et contrainte

Il est difficile d'obtenir des résultats propres car les dérivées en jeu ont des valeurs assez diverses et il difficile de trouver un pas de descente correct pour faire fonctionner l'algorithme d'Uzawa. Les essais ont donc été abandonnés ici au profit du Lagrangien Augmenté.

## Augmented Lagrangian Method

Les tests sont donc faits pour une optimisation simultanée de l'objectif de changement de phase et de réduction de la contrainte.

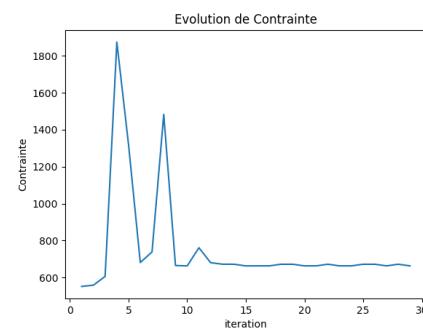
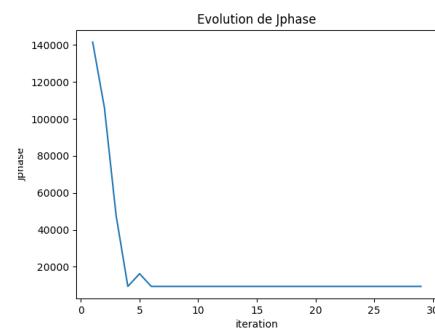
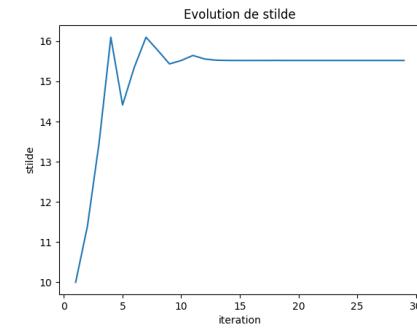
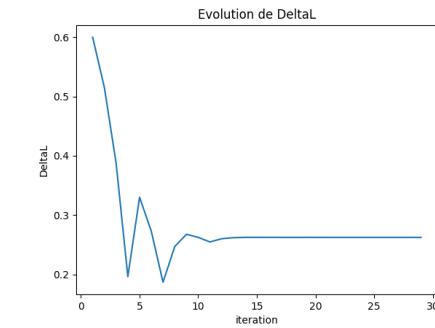
**Initialisation 1 :**

- valeurs initiales :  $\Delta L = 0.6$  et  $\tilde{s} = 10$
- paramètres d'optimisation :  $Lagini = 0$ ;  $\mu = 10$ ;  $tol_{sup} = 650$
- paramètres physiques :  $Tini = 30$ ;  $Tphase = 500$ ;  $Tsup = 750$  et  $epCaracCoucheCarre = 0.0001$ ;

Le résultat final est :

- $\Delta L = 0.26235$ ;  $\tilde{s} = 15.5214$
- $J_{phase} = 9391.72$ ;  $C = 663.527$

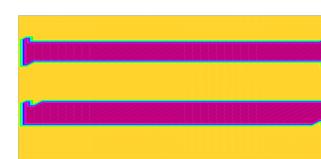
L'évolution en fonction des itérations de chaque grandeur est présentée dans les figures suivantes :



Iteration 1



Iteration 2



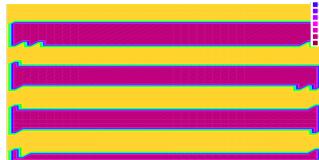
Iteration 3



Iteration 4



Iteration 5



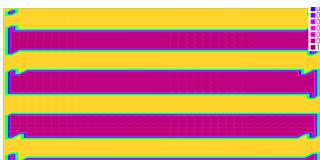
Iteration 6



Iteration 7



Iteration 8



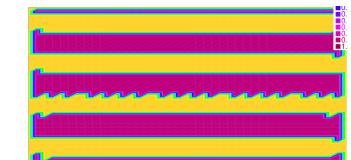
Iteration 9



Iteration 10



Iteration 11



Iteration 12



Iteration 13



Iteration 14



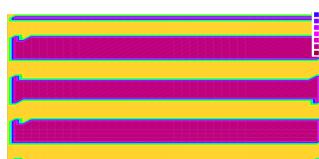
Iteration 15



Iteration 16



Iteration 17



Iteration 18



Iteration 19



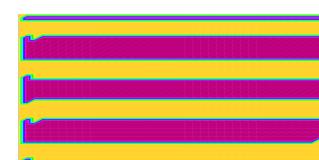
Iteration 20



Iteration 21



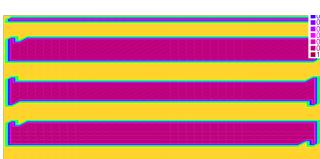
Iteration 22



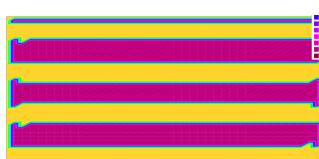
Iteration 23



Iteration 24



Iteration 25



Iteration 26



Iteration 27



Iteration 28

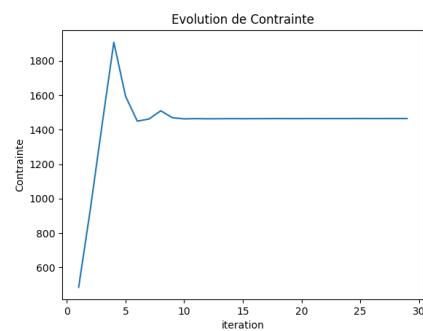
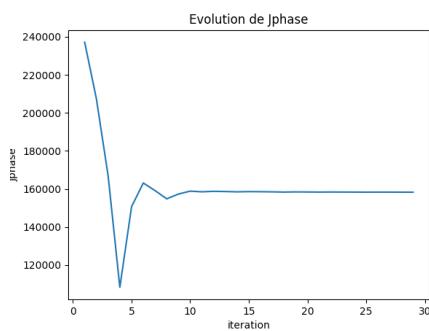
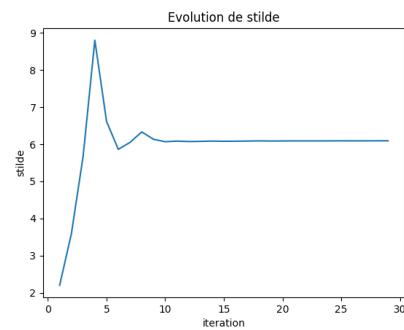
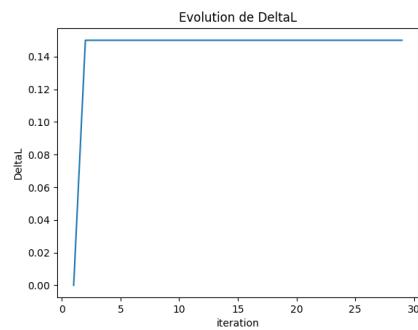
### Initialisation 2 :

- valeurs initiales :  $\Delta L = 0$  et  $\tilde{s} = 1.1 * L_l = 2.2$  avec  $L_l$  la longueur d'une ligne
- paramètres d'optimisation :  $Lagini = 0$ ;  $\mu = 10$ ;  $\text{tol}_{sup} = 650$
- paramètres physiques :  $Tini = 30$ ;  $Tphase = 500$ ;  $Tsup = 750$  et  $epCaracCoucheCarre = 0.0001$ ;

Le résultat final est :

- $\Delta L = 0.15$ ;  $\tilde{s} = 6.0934$
- $J_{phase} = 158287$ ;  $C = 1464.82$

Le résultat n'est pas terrible mais l'initialisation était probablement particulièrement non pertinente. L'évolution en fonction des itérations de chaque grandeur est présentée dans les figures suivantes :



Iteration 1



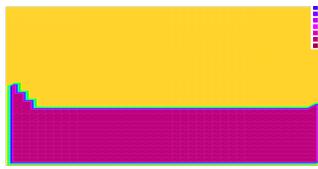
Iteration 2



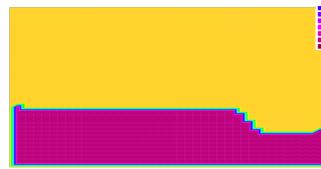
Iteration 3



Iteration 4



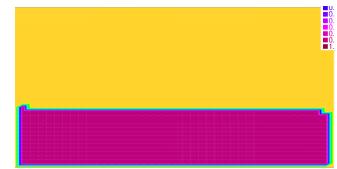
Iteration 5



Iteration 6



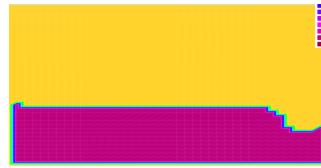
Iteration 7



Iteration 8



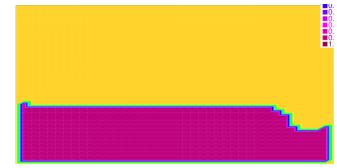
Iteration 9



Iteration 10



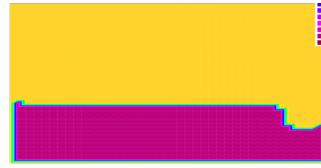
Iteration 11



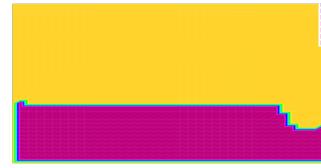
Iteration 12



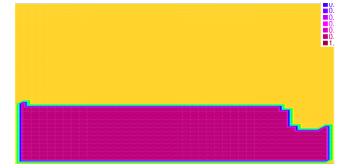
Iteration 13



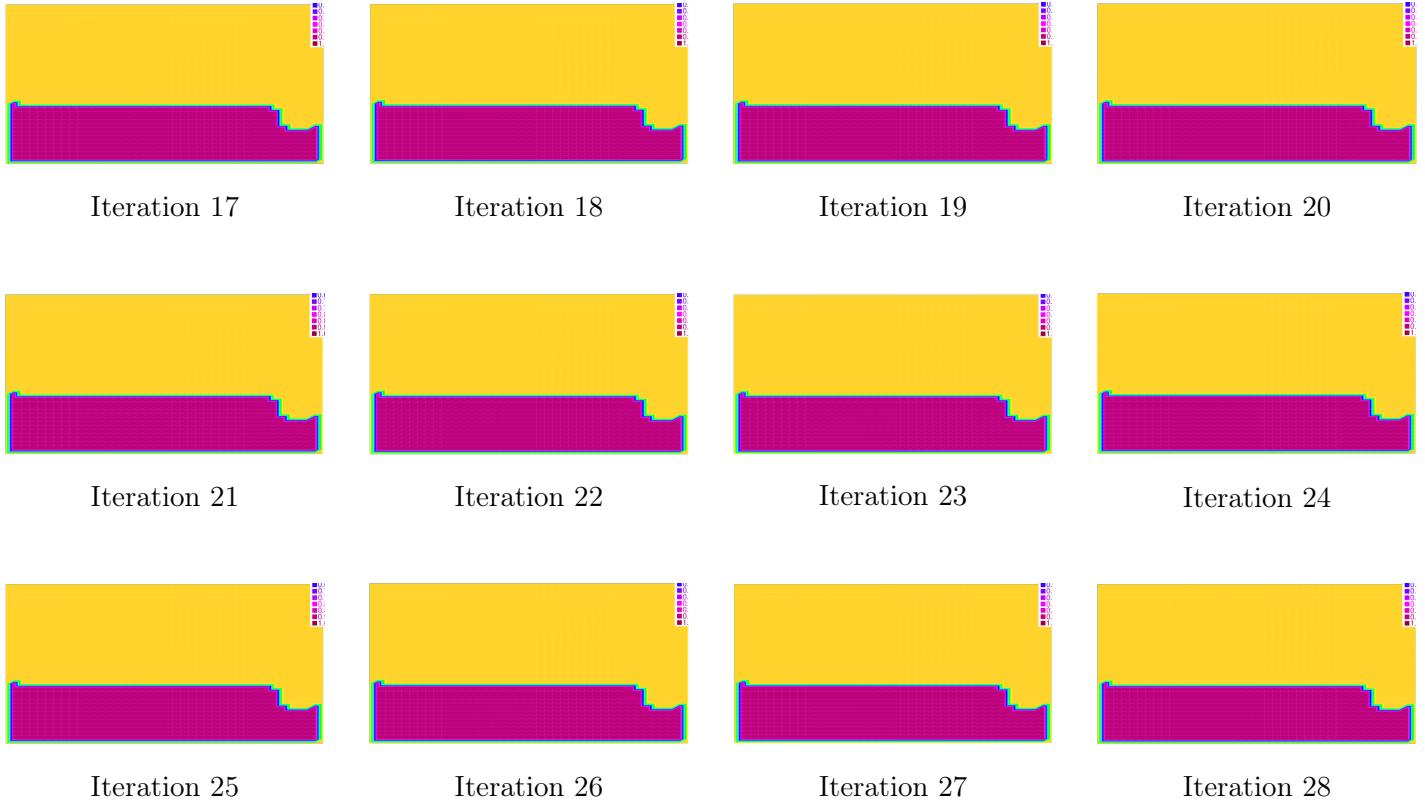
Iteration 14



Iteration 15



Iteration 16



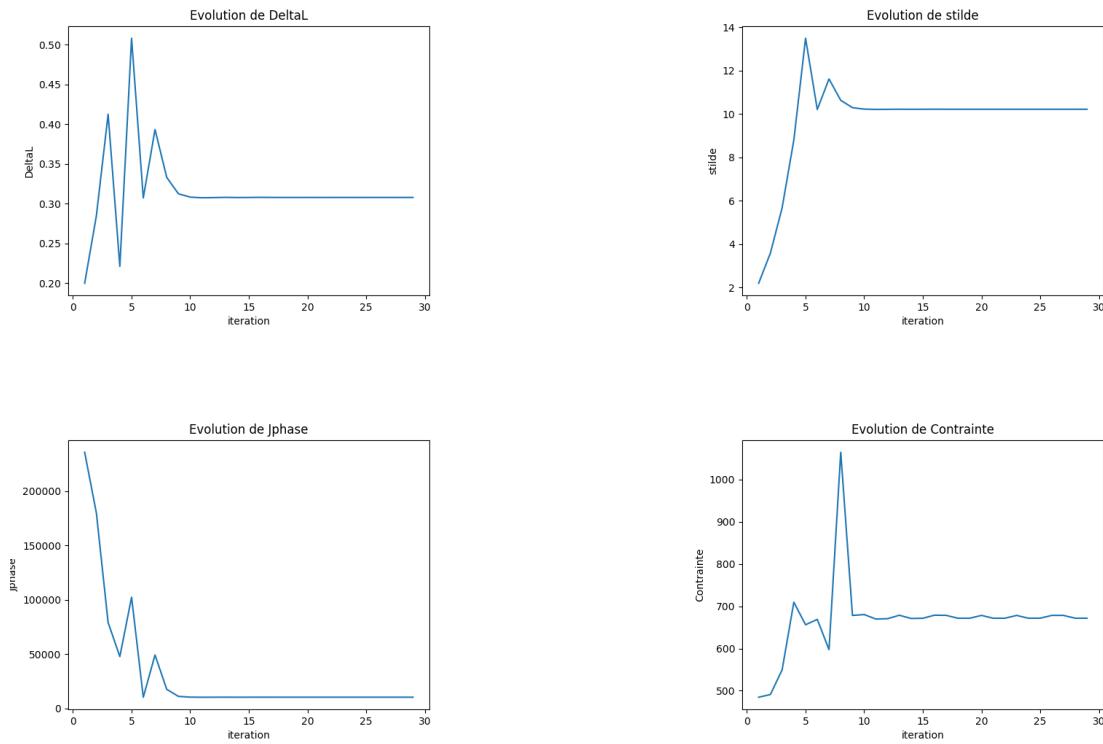
### Initialisation 3 :

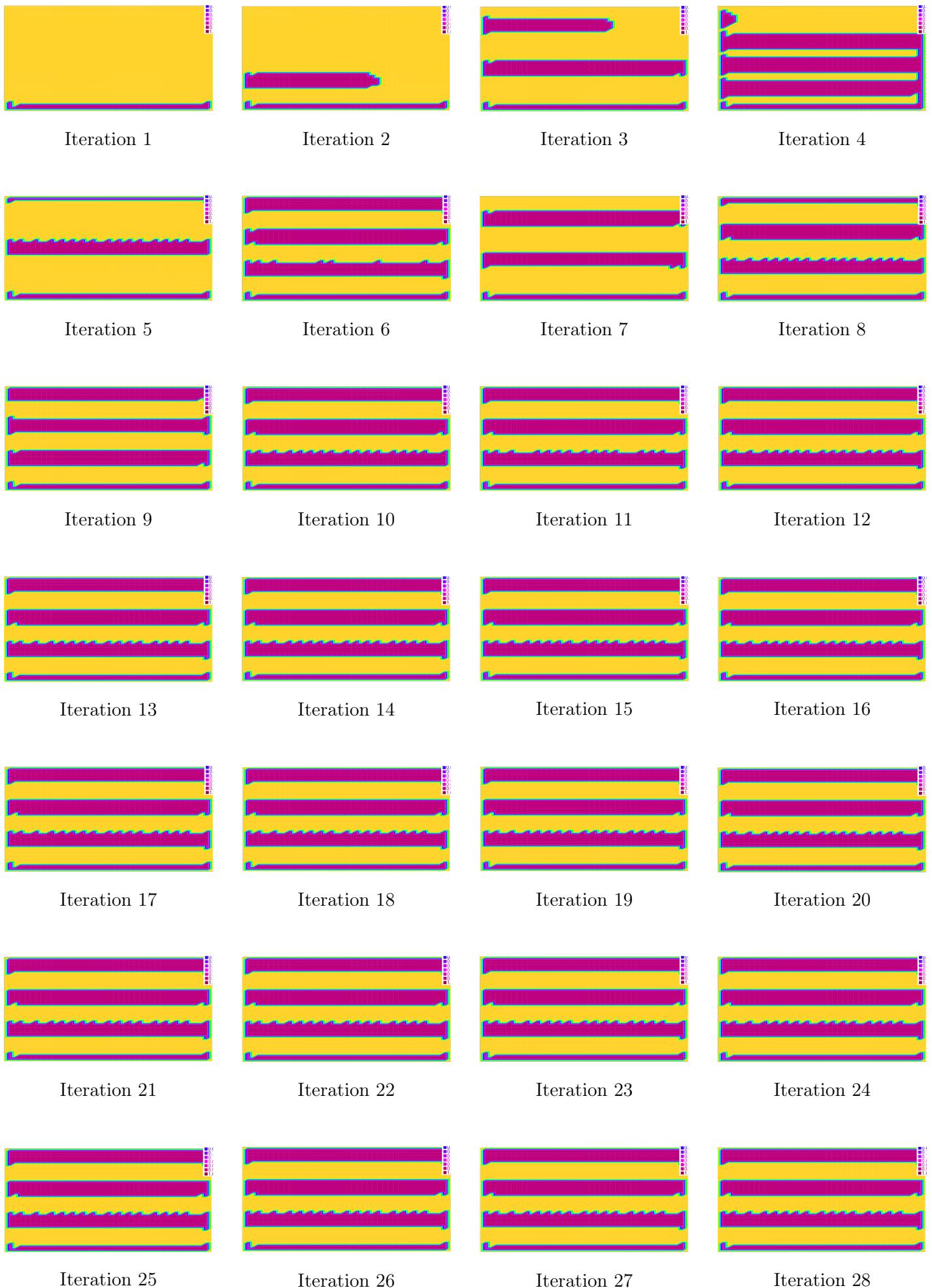
- valeurs initiales :  $\Delta L = 0.2$  et  $\tilde{s} = 1.1 * L_l = 2.2$  avec  $L_l$  la longueur d'une ligne
- paramètres d'optimisation :  $Lagini = 0$ ;  $\mu = 10$ ;  $tol_{sup} = 650$
- paramètres physiques :  $Tini = 30$ ;  $Tphase = 500$ ;  $Tsup = 750$  et  $epCaracCoucheCarre = 0.0001$ ;

Le résultat final est :

- $\Delta L = 0.307939$ ;  $\tilde{s} = 10.2199$
- $J_{phase} = 10254.4$ ;  $C = 671.638$

Le résultat est à peu près similaire à celui de l'initialisation 1. L'évolution en fonction des itérations de chaque grandeur est présentée dans les figures suivantes :



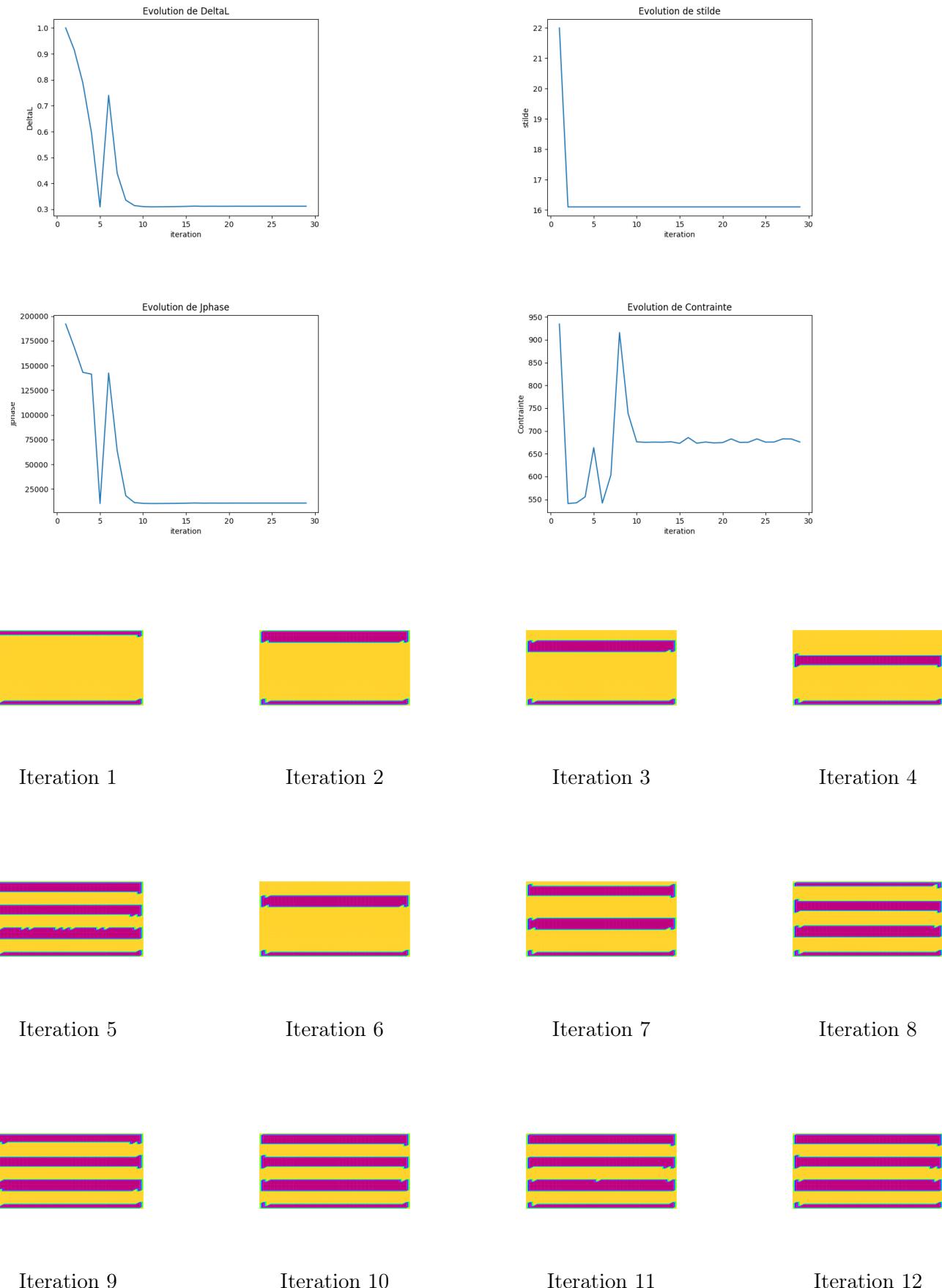


#### Initialisation 4 :

- valeurs initiales :  $\Delta L = \Delta L_{max} = 1$  et  $\tilde{s} = 11 * L_l = 22$  avec  $L_l$  la longueur d'une ligne

- paramètres d'optimisation :  $L_{ini} = 0$ ;  $\mu = 10$ ;  $tol_{sup} = 650$
  - paramètres physiques :  $T_{ini} = 30$ ;  $T_{phase} = 500$ ;  $T_{sup} = 750$  et  $epCaracCoucheCarre = 0.0001$ ;
- Le résultat final est :
- $\Delta L = 0.311886$ ;  $\tilde{s} = 16.0997$
  - $J_{phase} = 10874.6$ ;  $C = 675.787$

On retrouve là encore les résultats précédent. L'évolution en fonction des itérations de chaque grandeur est présentée dans les figures suivantes :





Iteration 13



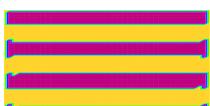
Iteration 14



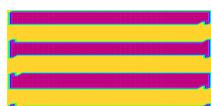
Iteration 15



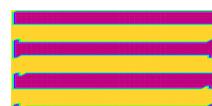
Iteration 16



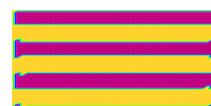
Iteration 17



Iteration 18



Iteration 19



Iteration 20



Iteration 21



Iteration 22



Iteration 23



Iteration 24



Iteration 25



Iteration 26



Iteration 27



Iteration 28

## Variation : lignes parcourues dans un sens puis dans l'autre

On s'intéresse maintenant au cas où, au lieu de recommencer toujours de la gauche les lignes droites, on alterne en recommençant du côté où l'on s'est arrêté. Il s'agit alors de reprogrammer la source et de déterminer sa dérivée.

On détermine l'évolution de l'abscisse et de l'ordonnée au cours du temps :

$$\begin{cases} \begin{cases} x_C(t) = x_{M0} + \left( t - \text{Ent}\left[\frac{t}{t_{Li}}\right] * t_{Li} \right) * V & \text{si } \text{Ent}\left[\frac{t}{t_{Li}}\right] \text{ pair} \\ x_C(t) = x_{M1} - \left( t - \text{Ent}\left[\frac{t}{t_{Li}}\right] * t_{Li} \right) * V & \text{si } \text{Ent}\left[\frac{t}{t_{Li}}\right] \text{ impair} \end{cases} \\ y_C(T) = \text{Ent}\left[\frac{t}{t_{Li}}\right] * \Delta L \end{cases} \quad (37)$$

On a alors

$$Q(t, x, y, \Delta L, \tilde{s}) = S(t, x, y, \Delta L) * h_\epsilon(\tilde{s} - tV) \quad (38)$$

## Résultats

Les initialisations 1 et 3 ont été testées dans ce cas.

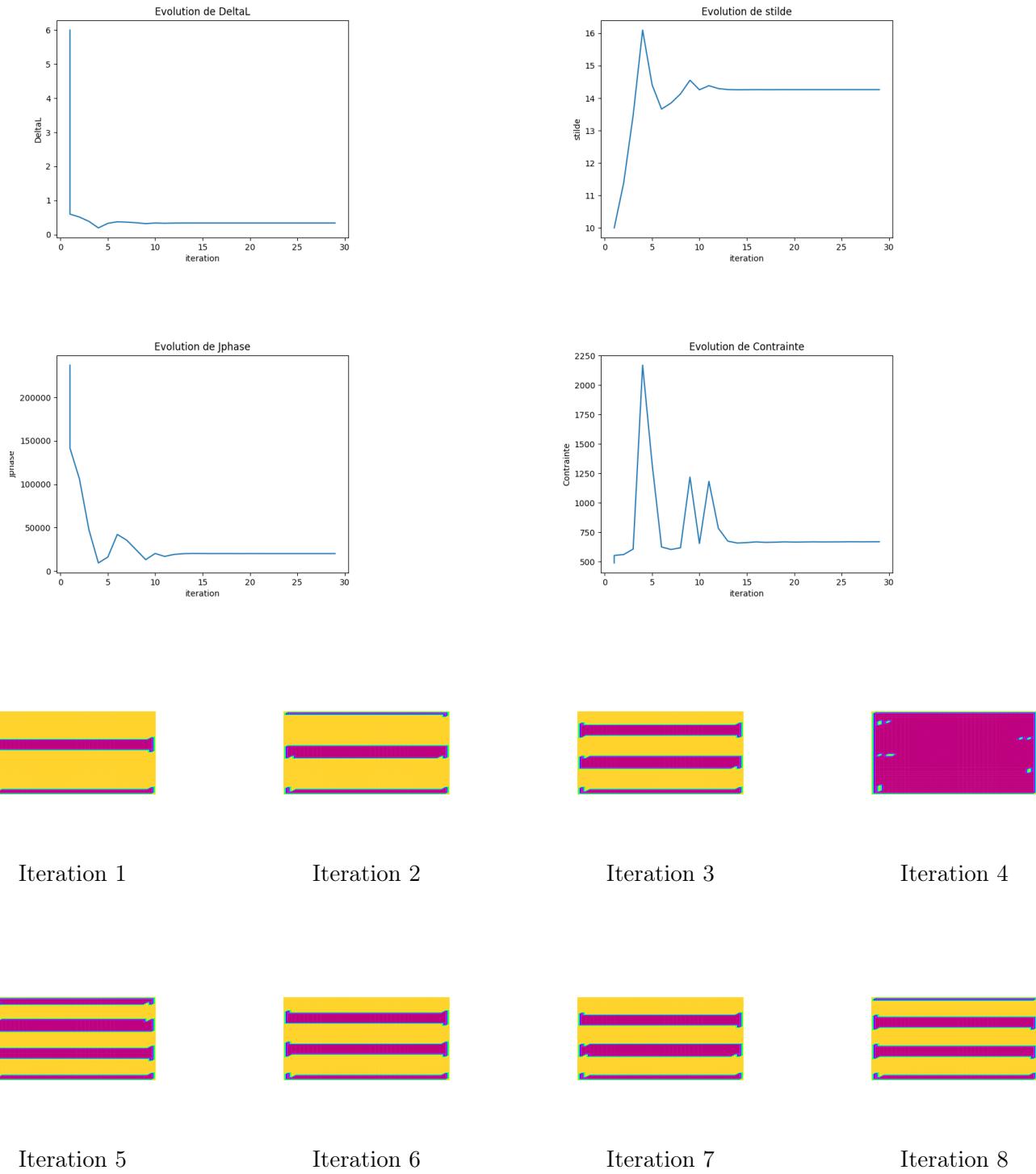
## Initialisation 1 :

- valeurs initiales :  $\Delta L = 0.6$  et  $\tilde{s} = 10$
- paramètres d'optimisation :  $Lag_{ini} = 0$ ;  $\mu = 10$ ;  $tol_{sup} = 650$
- paramètres physiques :  $T_{ini} = 30$ ;  $T_{phase} = 500$ ;  $T_{sup} = 750$  et  $epCaracCoucheCarre = 0.0001$ ;

Le résultat final est :

- $\Delta L = 0.339235$ ;  $\tilde{s} = 14.2642$
- $J_{phase} = 20179.4$ ;  $C = 668.705$

L'évolution en fonction des itérations de chaque grandeur est présentée dans les figures suivantes :





Iteration 9



Iteration 10



Iteration 11



Iteration 12



Iteration 13



Iteration 14



Iteration 15



Iteration 16



Iteration 17



Iteration 18



Iteration 19



Iteration 20



Iteration 21



Iteration 22



Iteration 23



Iteration 24



Iteration 25



Iteration 26



Iteration 27



Iteration 28

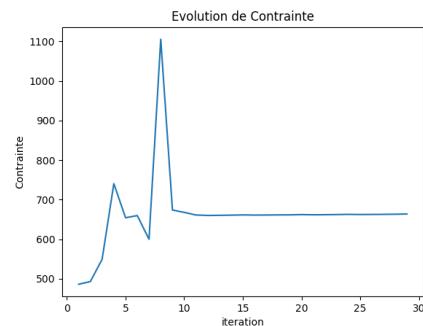
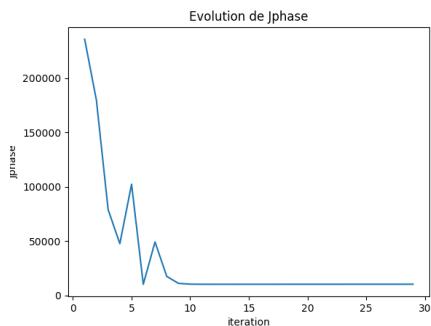
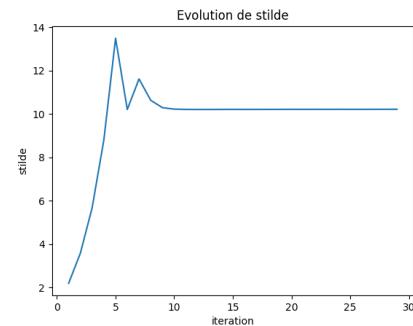
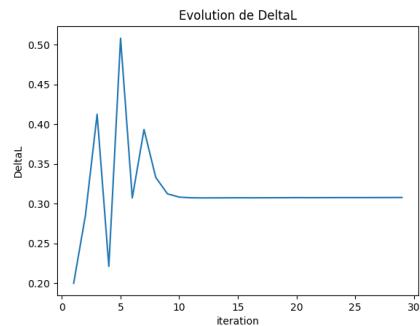
### Initialisation 3 :

- valeurs initiales :  $\Delta L = 0.2$  et  $\tilde{s} = 1.1 * L_l = 2.2$  avec  $L_l$  la longueur d'une ligne
- paramètres d'optimisation :  $Lagini = 0$ ;  $\mu = 10$ ;  $tol_{sup} = 650$
- paramètres physiques :  $Tini = 30$ ;  $Tphase = 500$ ;  $Tsup = 750$  et  $epCaracCoucheCarre = 0.0001$ ;

Le résultat final est :

- $\Delta L = 0.30789$ ;  $\tilde{s} = 10.2191$
- $J_{phase} = 10243.9$ ;  $C = 663.754$

Le résultat est à peu près similaire à celui de l'initialisation 1. L'évolution en fonction des itérations de chaque grandeur est présentée dans les figures suivantes :



Iteration 1



Iteration 2



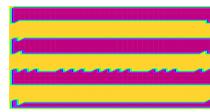
Iteration 3



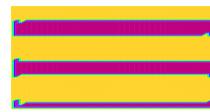
Iteration 4



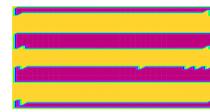
Iteration 5



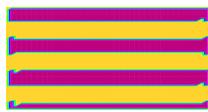
Iteration 6



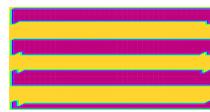
Iteration 7



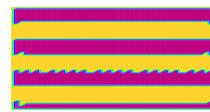
Iteration 8



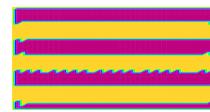
Iteration 9



Iteration 10



Iteration 11



Iteration 12



Iteration 13



Iteration 14



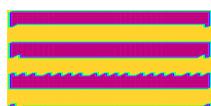
Iteration 15



Iteration 16



Iteration 17



Iteration 18



Iteration 19



Iteration 20



Iteration 21



Iteration 22



Iteration 23



Iteration 24



Iteration 25



Iteration 26



Iteration 27



Iteration 28

## Pour la suite

Plusieurs pistes pour continuer ce projet sont possibles pour la suite :

- Optimisation de la distance entre les lignes pour une trajectoire en lignes droites sur une forme rectangulaire :
  - Amélioration des données et de la définition des fonctions objectifs et de contraintes pour être au plus près du problème physique.
  - Traitement de la contrainte de manière plus forte ou algorithme d'optimisation qui converge plus rapidement. (à voir pour ce point là)
- Optimisation de la distance entre les lignes pour une trajectoire en lignes droites :
  - Adaptation à d'autres formes :
    - \* formes dont on connaît le paramétrage des bords : triangle, ovale, cercle, ... Le problème principal à régler est que la longueur de chaque ligne dépend de  $\Delta L$  et on induit de nouvelles discontinuités beaucoup plus compliquées à régler dans la source.
    - \* formes dont on n'a pas d'équation explicite (levelset de fonction, ...)
- Trajectoires en lignes droite : ajout d'un angle de parcours et optimisation de l'angle.

- en 2D : on le fait pour une seule couche afin de voir si physiquement, ça fait une différence (ça en fait une d'un point de vue vitesse de parcours, cf Rajan 2001)
- en 3D : on construit couche après couche et on veut savoir l'évolution de la distance entre les lignes ainsi que de l'angle pour chaque couche.

- Adaptation à d'autres types de trajectoires :

- zig-zag
- isovaleurs
- ... mais ce qui est sûr, c'est qu'il faudra trouver un moyen de gérer la discontinuité de la source.