

# Méthode 4

## Principe

- Il s'agit de la méthode 2 avec l'algorithme du gradient avec pas optimal expliqué dans les diapos suivante.

## Comment s'y référer et l'exécuter??

Se référer au code python 'Méthode 4' sur cette page :

<https://github.com/ComeLassarat/PETT/blob/master/Livrables/Tache5-4/M%C3%A9thode%204.py>

Il suffit d'exécuter le code pour obtenir les courbes:

- celle donnant l'erreur en fonction du nombre d'itération
- celle donnant la trajectoire pointée et celle simulée qui correspond au mieux à celle pointée dans le plan 2D de la caméra
- si les courbes semblent incorrectes ou ne s'affichent pas, il suffit de relancer le code en ayant commenté la ligne 446.

E Au hasard

Trajectoire simulée n1

Echantillonnage

Trajectoire simulée n2

Changement repère

Trajectoire simulée n2

Trajectoire pointée n2

n1=601 points

n2=14 points

On réalise un échantillonnage  
pour limiter les calculs

Espace 3D de simulation théorique

Espace 2D de la caméra

Evaluation erreur  
Position Initiale EPI

Tant que  $EPI > \text{seuil1}$

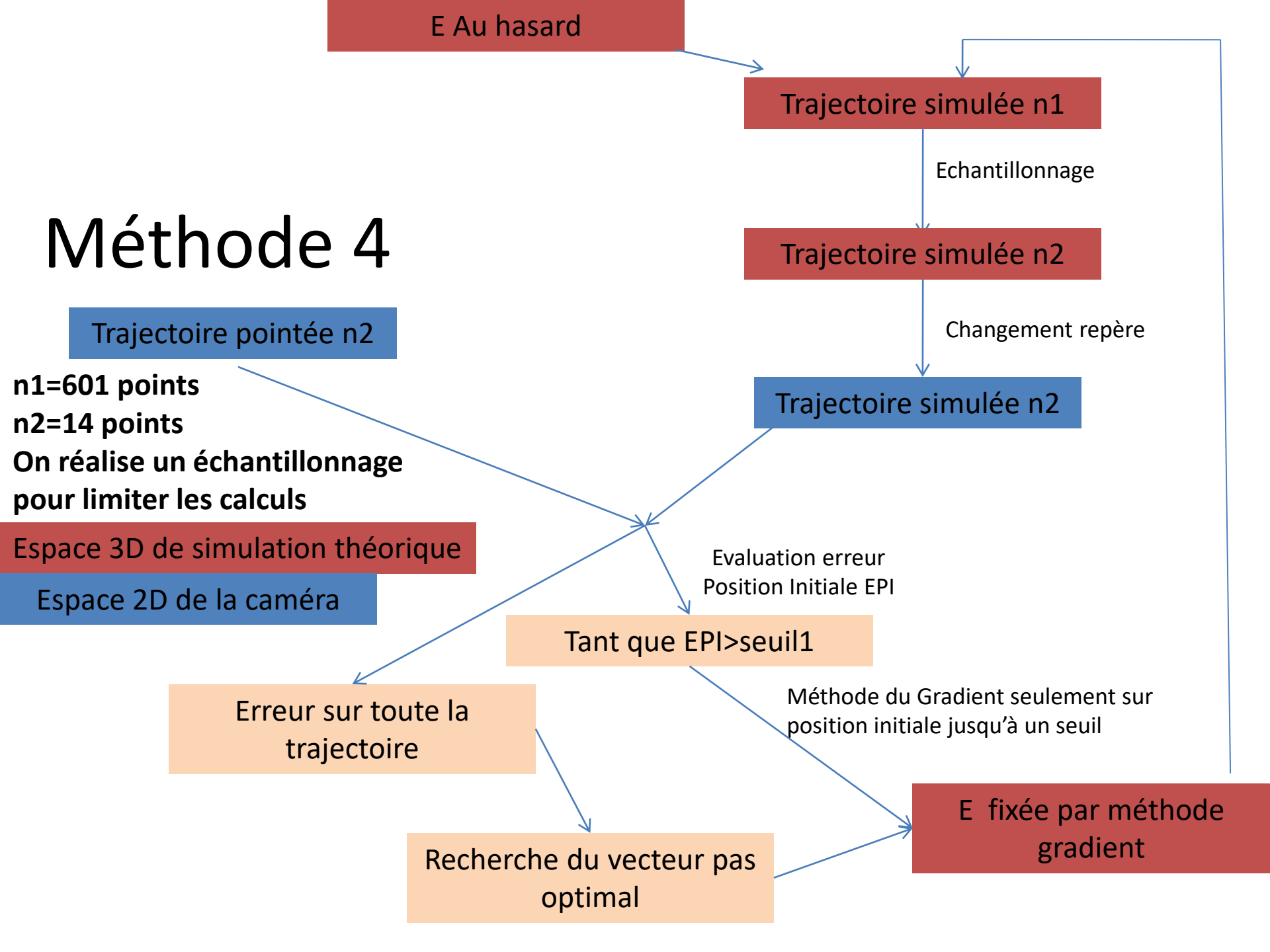
Erreur sur toute la  
trajectoire

Méthode du Gradient seulement sur  
position initiale jusqu'à un seuil

Recherche du vecteur pas  
optimal

E fixée par méthode  
gradient

# Méthode 4



# Méthode du Gradient Optimal

## Rappel des choix d'optimisation de la méthode 2:

Par rapport à la méthode 2, la méthode de minimisation de l'erreur est modifiée.  
En effet lors de l'itération principale de l'algorithme (qui est:  $\overrightarrow{E_{n+1}} = \overrightarrow{E_n} - d \cdot \overrightarrow{\nabla F}$  )  
on avait le système suivant:

$$\begin{pmatrix} X_o \\ Y_o \\ Z_o \\ V_{xo} \\ V_{yo} \\ V_{zo} \\ \omega_{xo} \\ \omega_{yo} \\ \omega_{zo} \end{pmatrix}_{n+1} = \begin{pmatrix} X_o \\ Y_o \\ Z_o \\ V_{xo} \\ V_{yo} \\ V_{zo} \\ \omega_{xo} \\ \omega_{yo} \\ \omega_{zo} \end{pmatrix}_n - d \cdot \begin{pmatrix} 0 \\ 0 \\ 0 \\ (\nabla F)_{v_x} \\ (\nabla F)_{v_y} \\ (\nabla F)_{v_z} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

On modifiait soit la vitesse initiale (comme ci-dessus) soit la position initiale soit l'effet initial avec un pas d constant (dans la méthode 2, on effectuait déjà un choix de pas (voir la fonction choixdelta) en fonction de la valeur de l'erreur: on affinait la recherche de la vitesse initiale avec un d petit quand l'erreur devenait petite).

Présentation des nouveaux choix d'optimisation:

La minimisation de l'erreur est modifiée pour deux raisons.  
D'une part on ne va plus scinder le problème de minimisation entre l'effet, la vitesse et la position mais on va traiter chacune des 9 coordonnées à part. On n'a donc pas une constante d comme précédemment mais une matrice.  
D'autre part le pas est choisi selon un modèle mathématique: **la méthode du gradient à pas optimal**.

On conserve le traitement de la position puis de la vitesse puis de l'effet comme dans la méthode 2 mais on choisit le vecteur pas  $\vec{R}$  de l'équation  $\vec{E}_{n+1} = \vec{E}_n - \vec{R} \cdot \nabla F$  pour chacune de ces étapes en trouvant le pas optimal (dans le cas de la vitesse par exemple ci-dessous en choisissant  $\alpha$ ,  $\beta$  et  $\gamma$ ).

$$\begin{pmatrix} X_o \\ Y_o \\ Z_o \\ V_{xo} \\ V_{yo} \\ V_{zo} \\ \omega_{xo} \\ \omega_{yo} \\ \omega_{zo} \end{pmatrix}_{n+1} = \begin{pmatrix} X_o \\ Y_o \\ Z_o \\ V_{xo} \\ V_{yo} \\ V_{zo} \\ \omega_{xo} \\ \omega_{yo} \\ \omega_{zo} \end{pmatrix}_n - \begin{pmatrix} 0 \\ 0 \\ 0 \\ \alpha \\ \beta \\ \gamma \\ 0 \\ 0 \\ 0 \end{pmatrix}^T \cdot \begin{pmatrix} 0 \\ 0 \\ 0 \\ (\nabla F)_{v_x} \\ (\nabla F)_{v_y} \\ (\nabla F)_{v_z} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Pour plus de détails sur la méthode du gradient à pas optimal, voir le cours d'Analyse Numérique (P21) de M. Vial et M. Seppecher.

# Présentation des Résultats

Cette nouvelle méthode permet, contrairement aux méthodes précédentes de réduire l'erreur de manière importante.

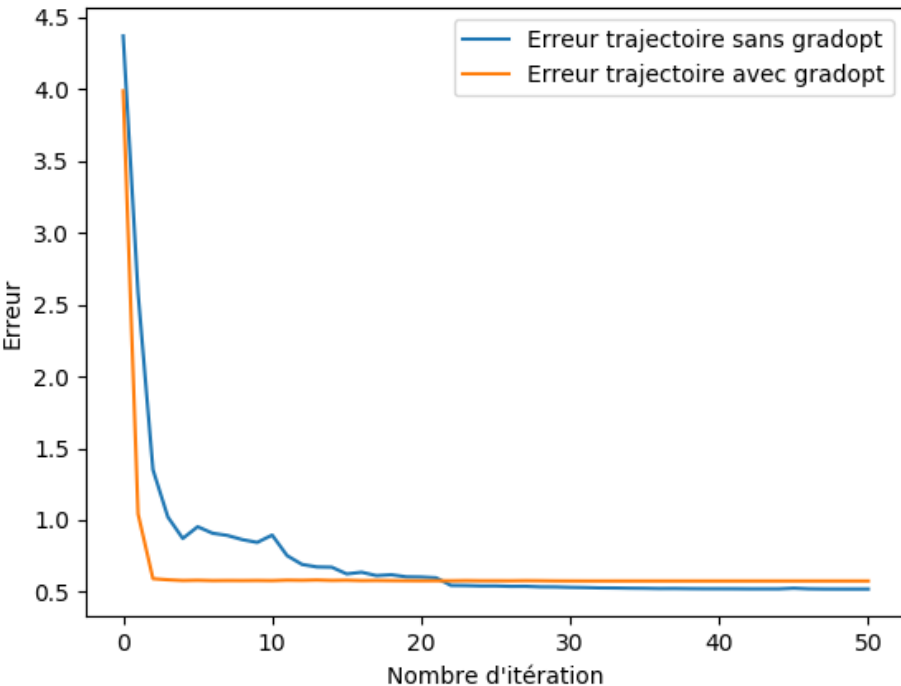
Ce résultat encourageant a un coût: la lenteur. En effet, pour calculer le pas optimal, on réalise beaucoup de calculs. Bien qu'on converge en un nombre d'itération réduit, ces itérations sont beaucoup plus longue.

# Evolution de l'erreur

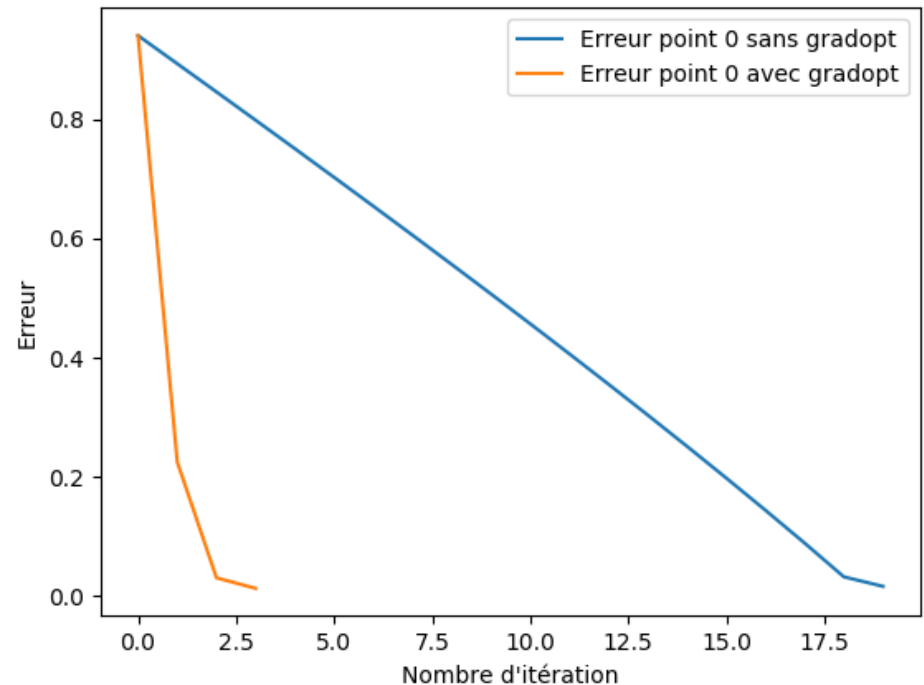
Ci-dessous sont visibles des comparaisons d'évolution de l'erreur en fonction du nombre d'itération pour la recherche du point initial et de la vitesse initial.

Sur chaque graphe est visible l'erreur en question avec la nouvelle méthode d'optimisation du pas ('erreur avec gradopt') et celle avec la méthode 2 ('erreur sans gradopt') pour un même vecteur de recherche initial.

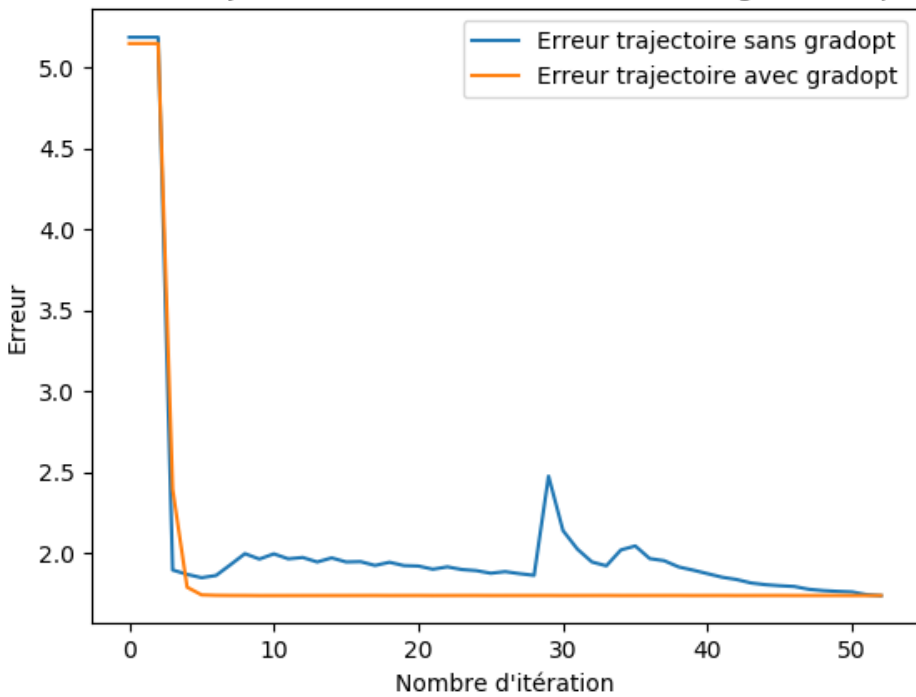
Erreur sur trajectoire avec et sans la méthode du gradient optimal



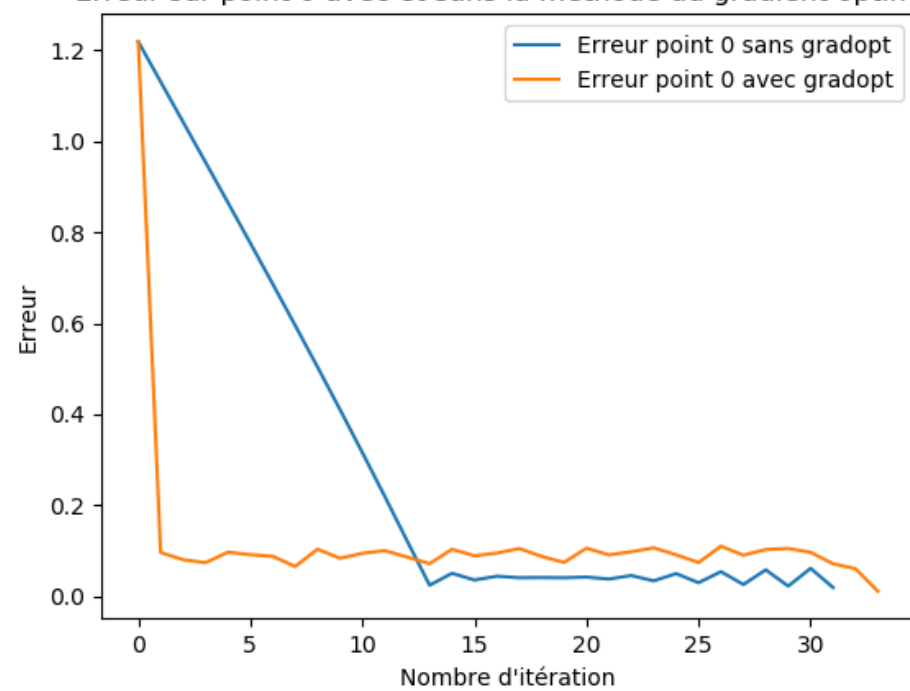
Erreur sur point 0 avec et sans la méthode du gradient optimal



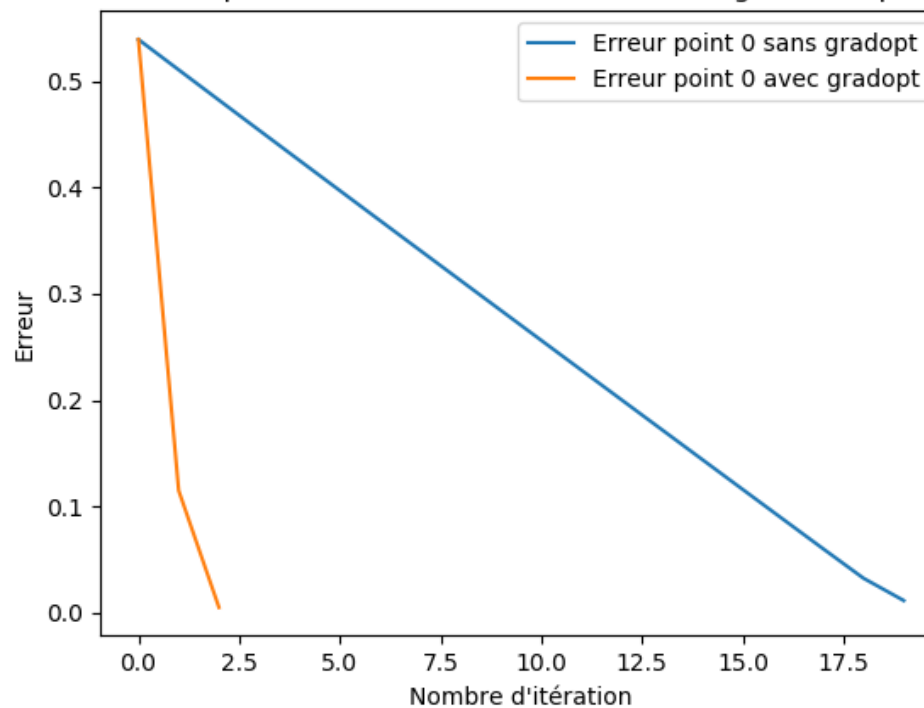
Erreur sur trajectoire avec et sans la méthode du gradient optimal



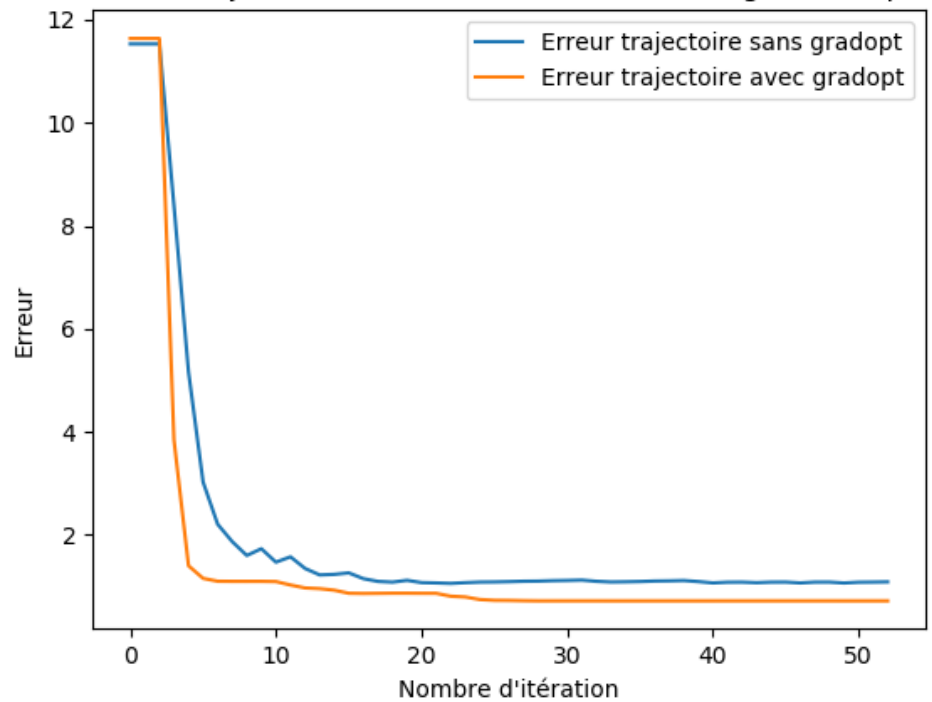
Erreur sur point 0 avec et sans la méthode du gradient optimal



Erreur sur point 0 avec et sans la méthode du gradient optimal



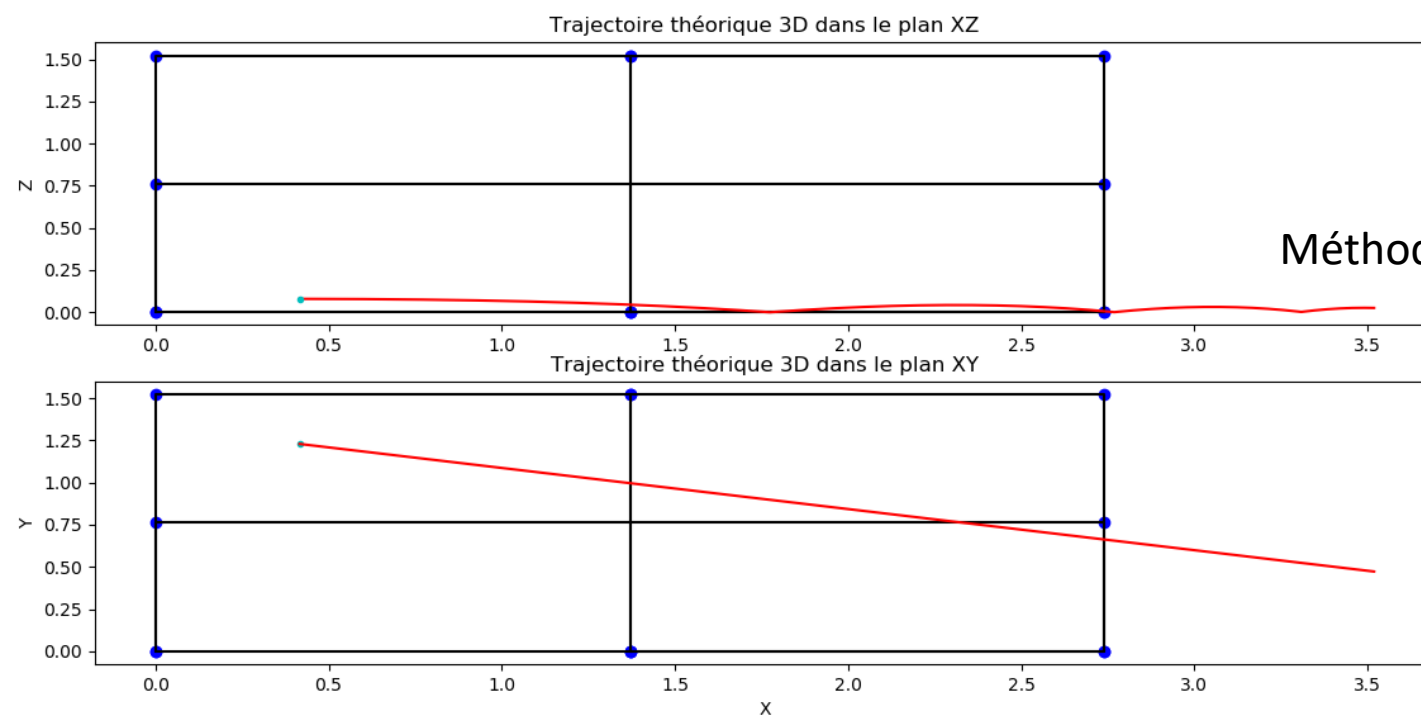
Erreur sur trajectoire avec et sans la méthode du gradient optimal



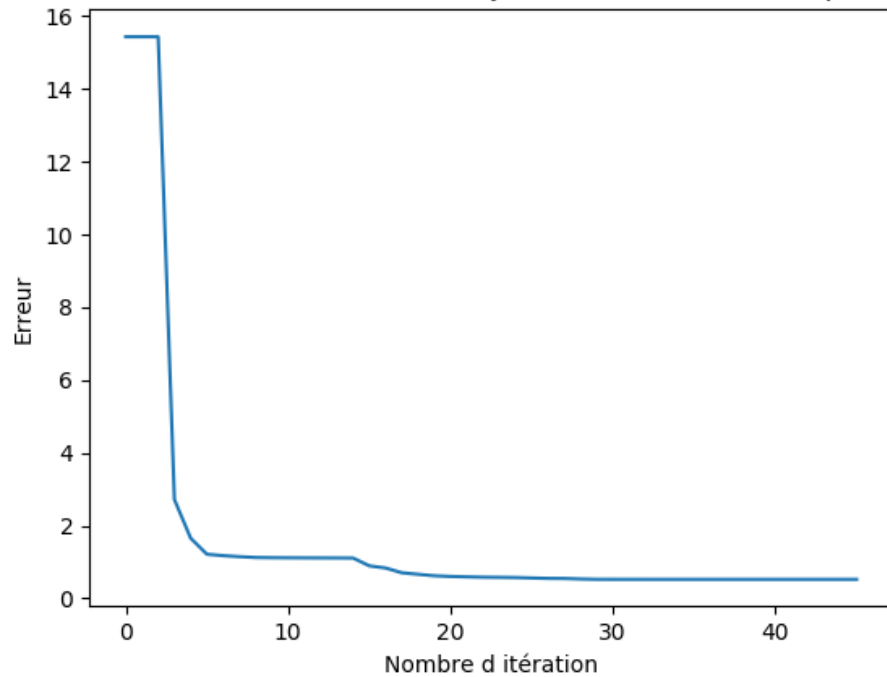
On constate que la différence est plus flagrante pour la recherche du point initial. En effet dans le cadre de la recherche de la vitesse initiale, on effectuait déjà une optimisation du pas (cependant pas pour chaque coordonnée mais pour les 3 coordonnées de vitesse en même temps) à chaque itération en fonction de la valeur de l'erreur. Pour la recherche du point initial, on n'a pas de saturation en erreur car passé un seuil, on passe sur la recherche de la vitesse initiale.



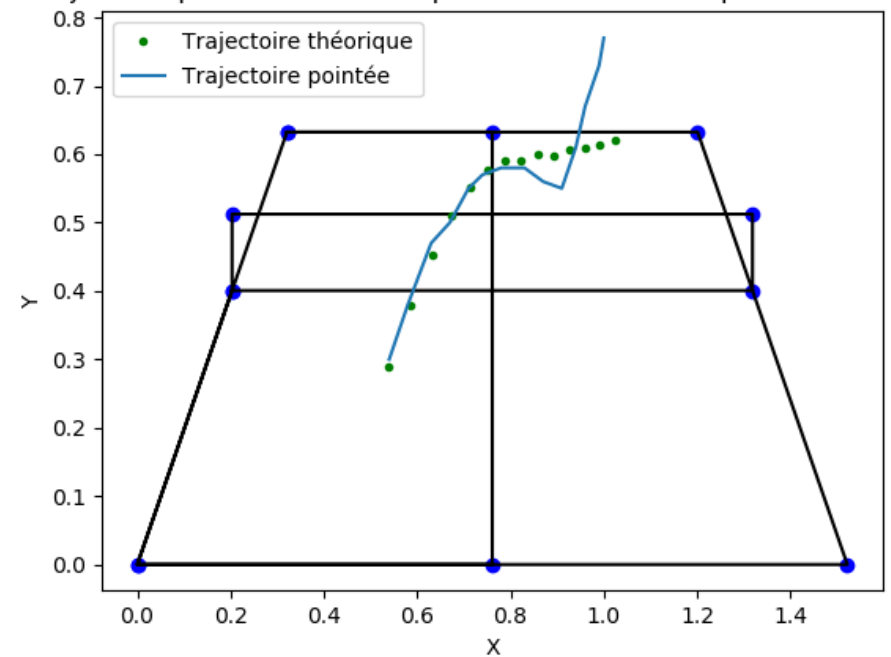
## Méthode 4: Coup 1



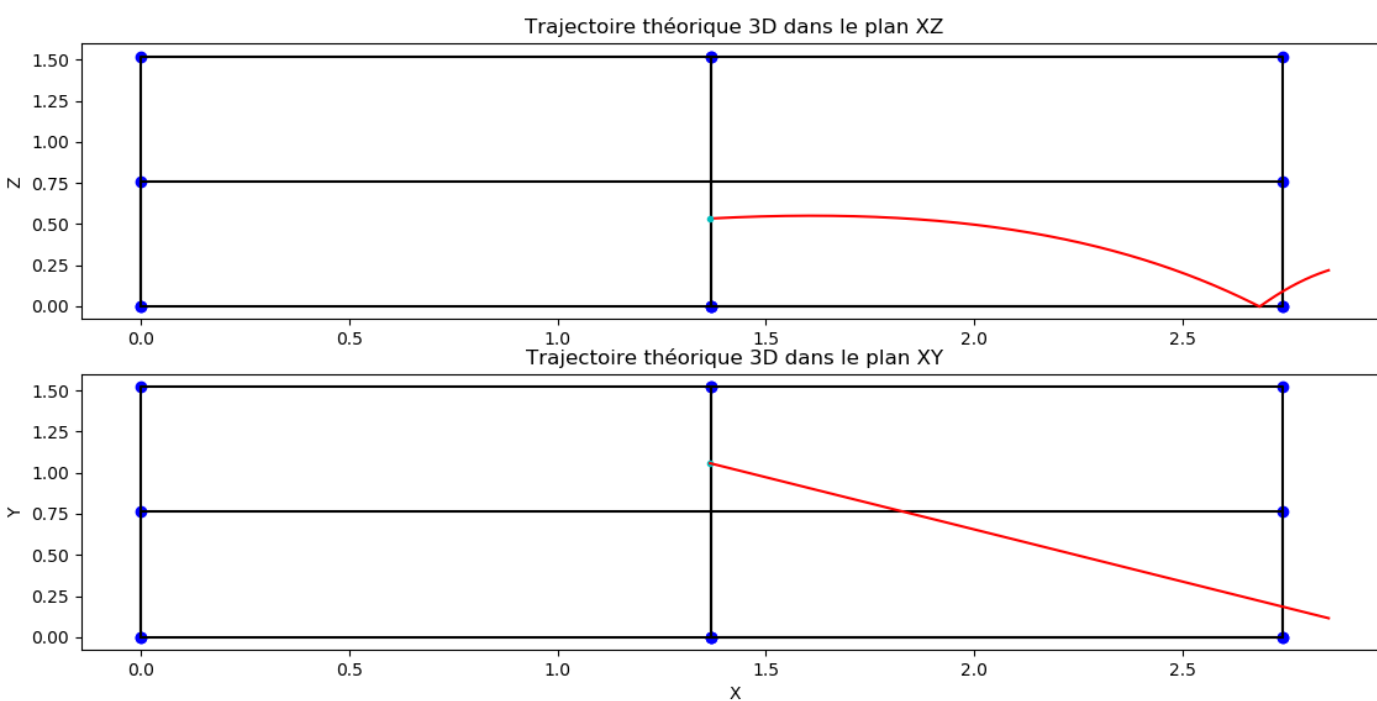
Evolution de l'erreur entre la trajectoire simulée et celle pointée



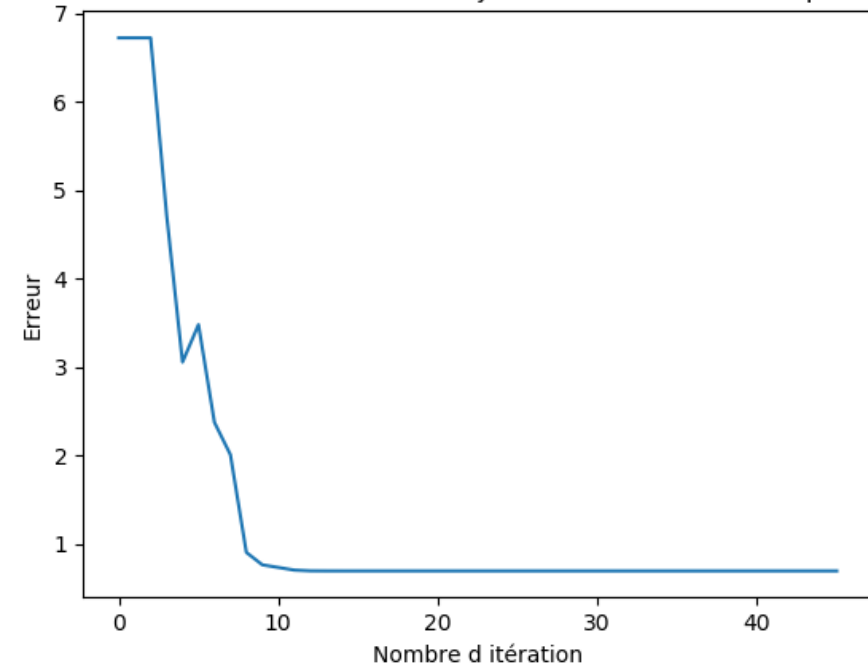
Trajectoire pointée et simulée par la théorie dans le plan de la caméra



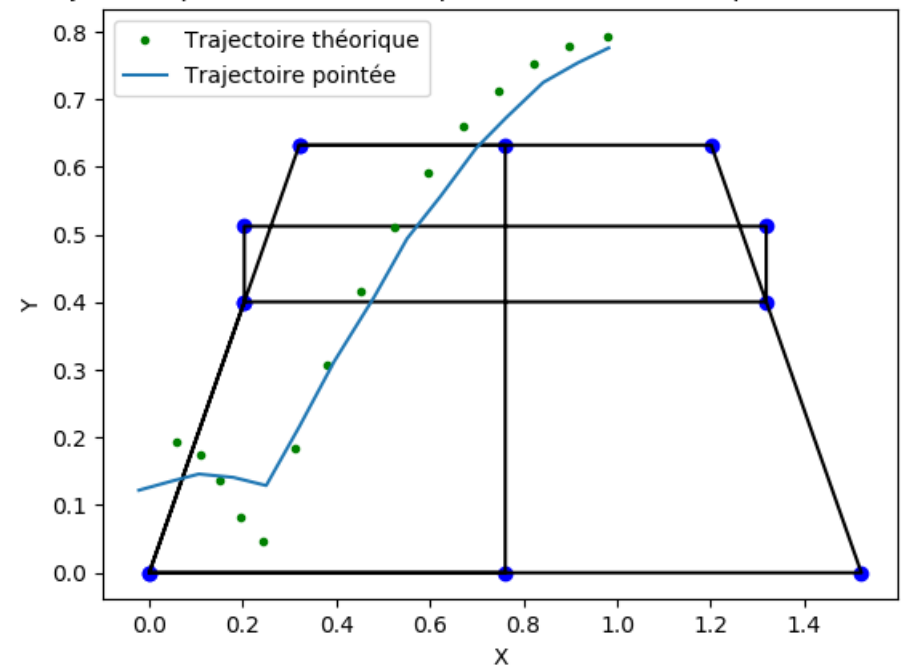
## Méthode 4: Coup 2



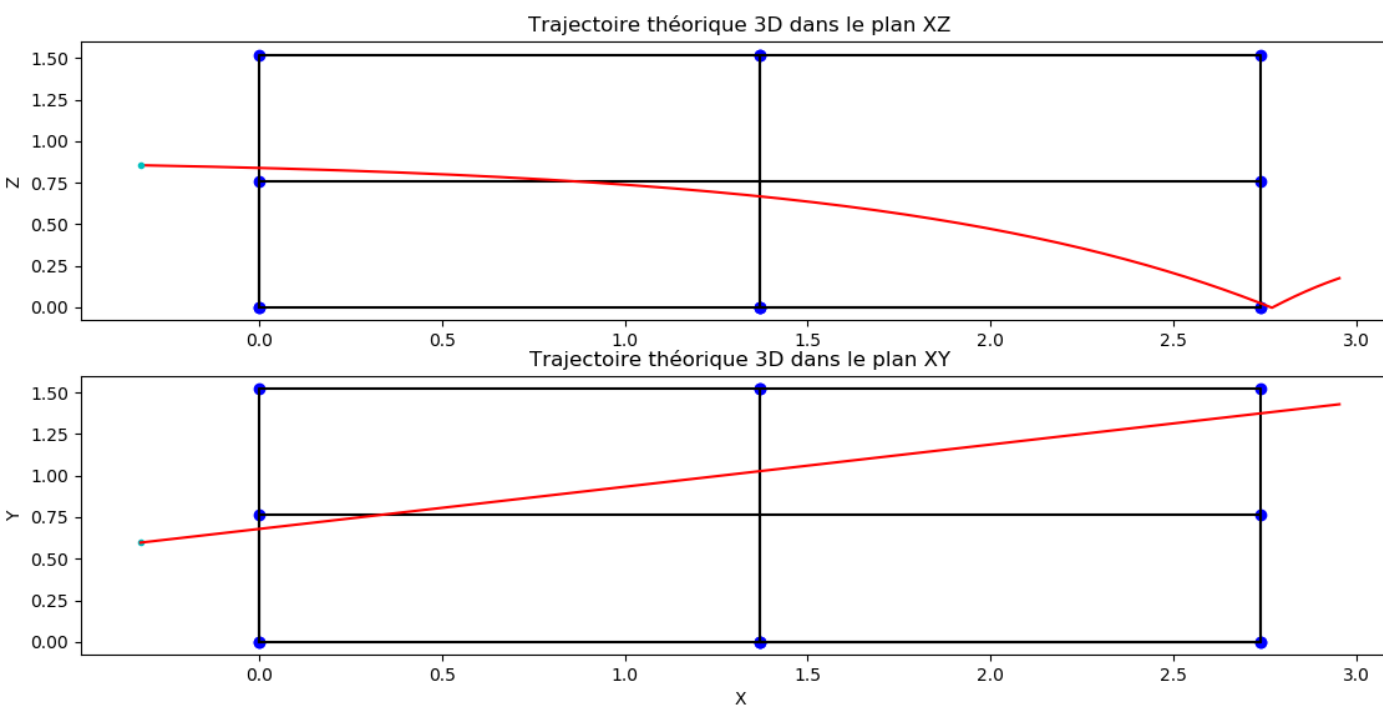
Evolution de l'erreur entre la trajectoire simulée et celle pointée



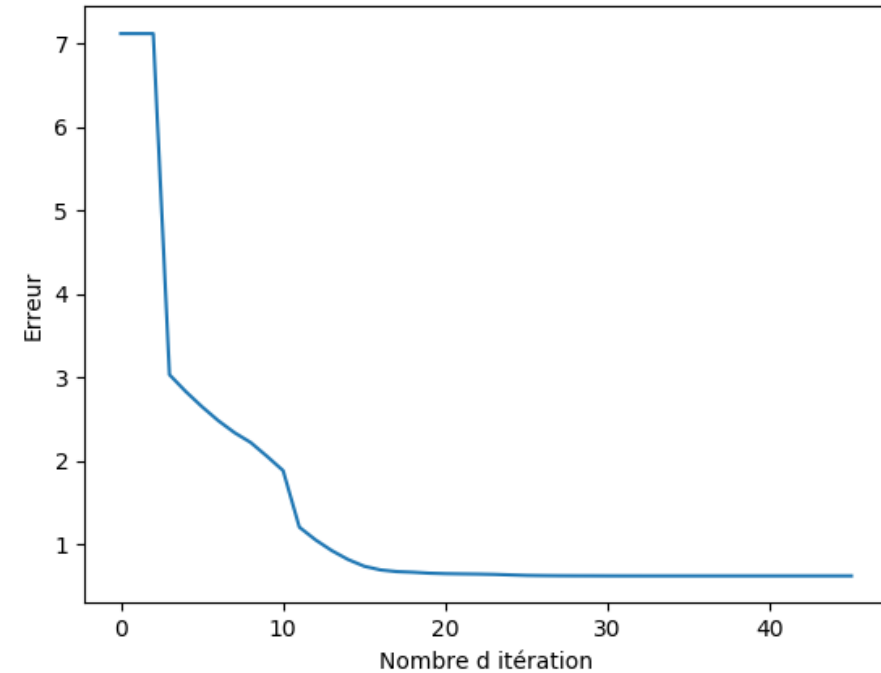
Trajectoire pointée et simulée par la théorie dans le plan de la caméra



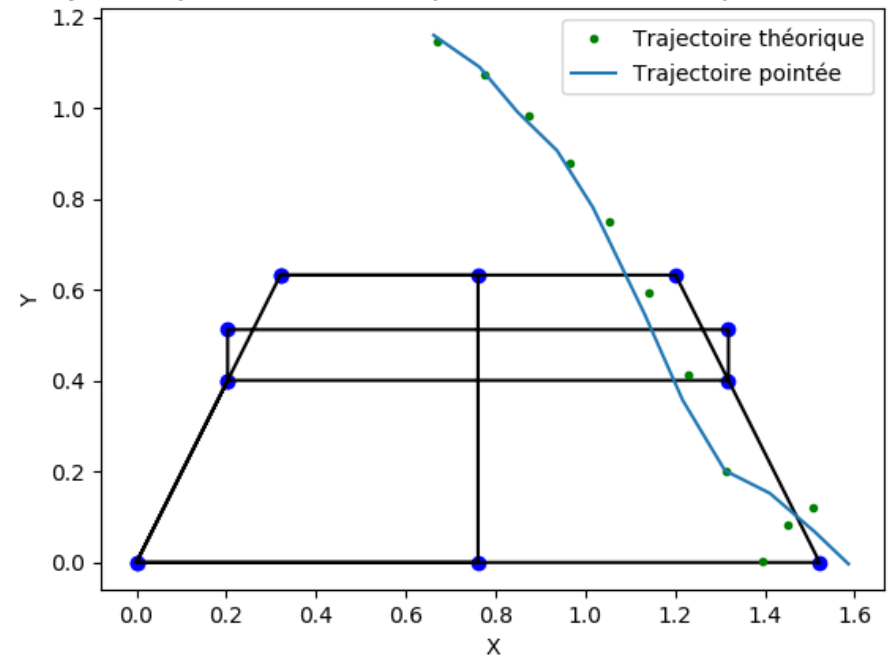
## Méthode 4: Coup 3



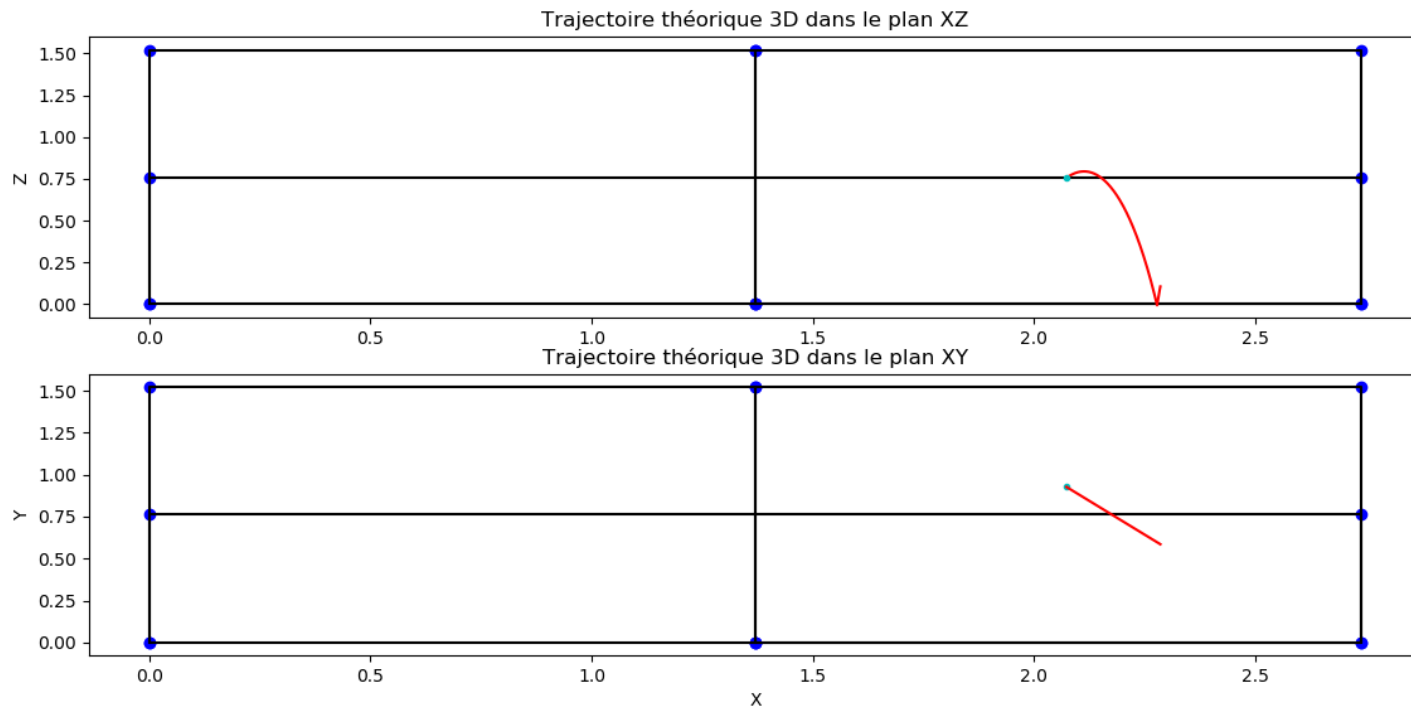
Evolution de l'erreur entre la trajectoire simulée et celle pointée



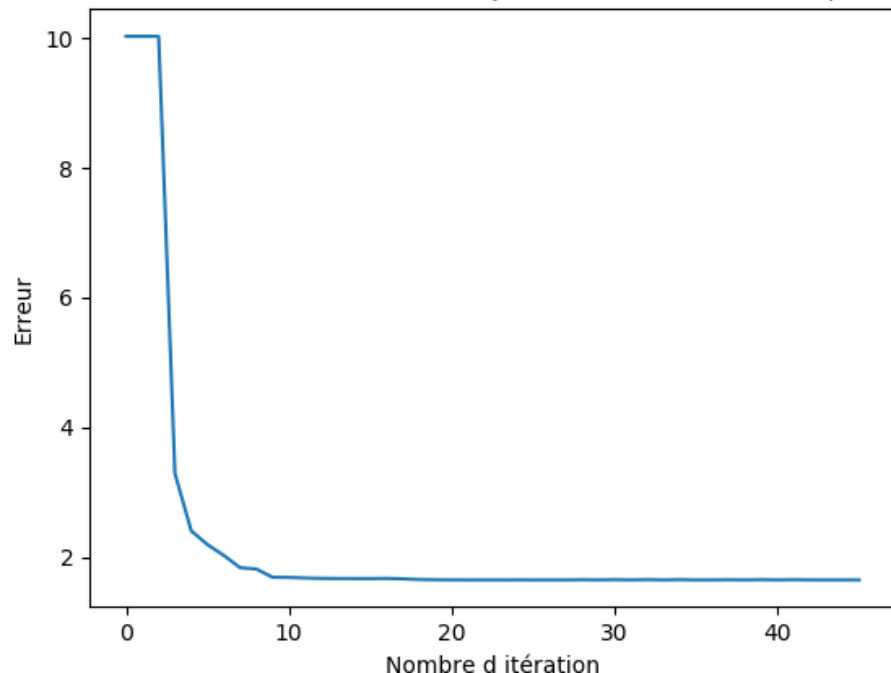
Trajectoire pointée et simulée par la théorie dans le plan de la caméra



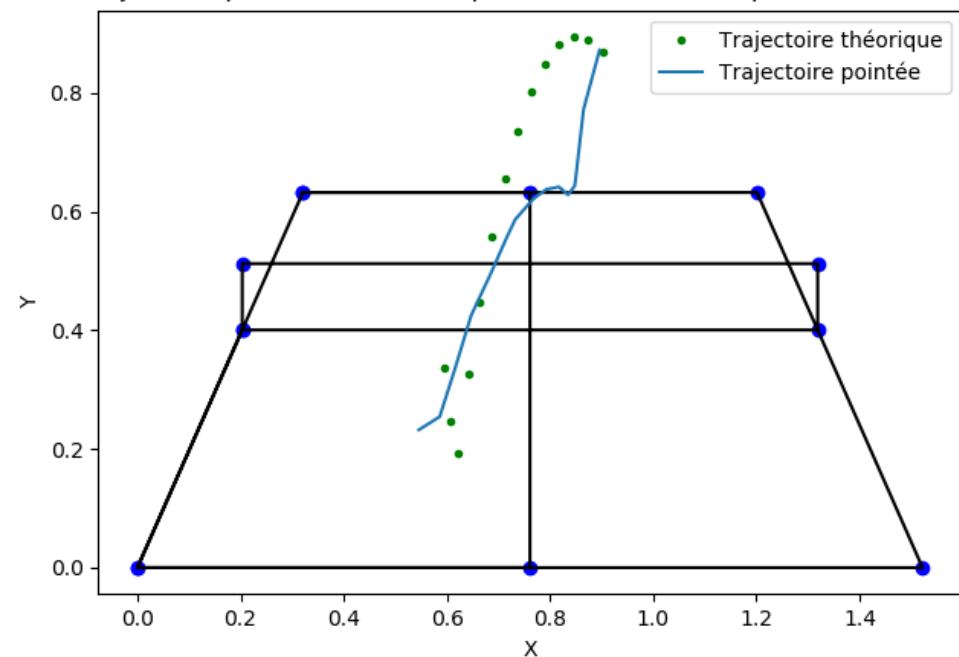
## Méthode 4: Service



Evolution de l'erreur entre la trajectoire simulée et celle pointée

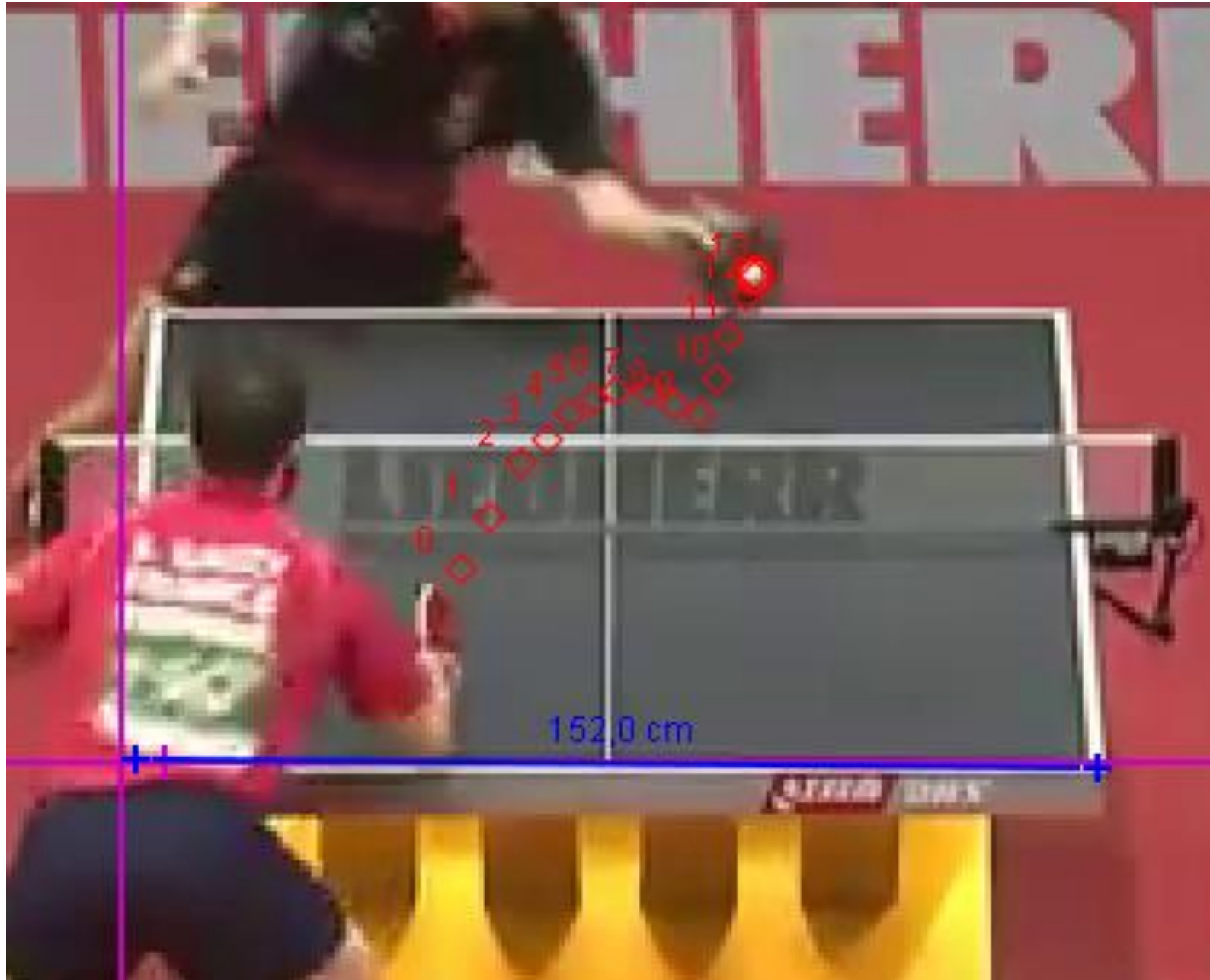


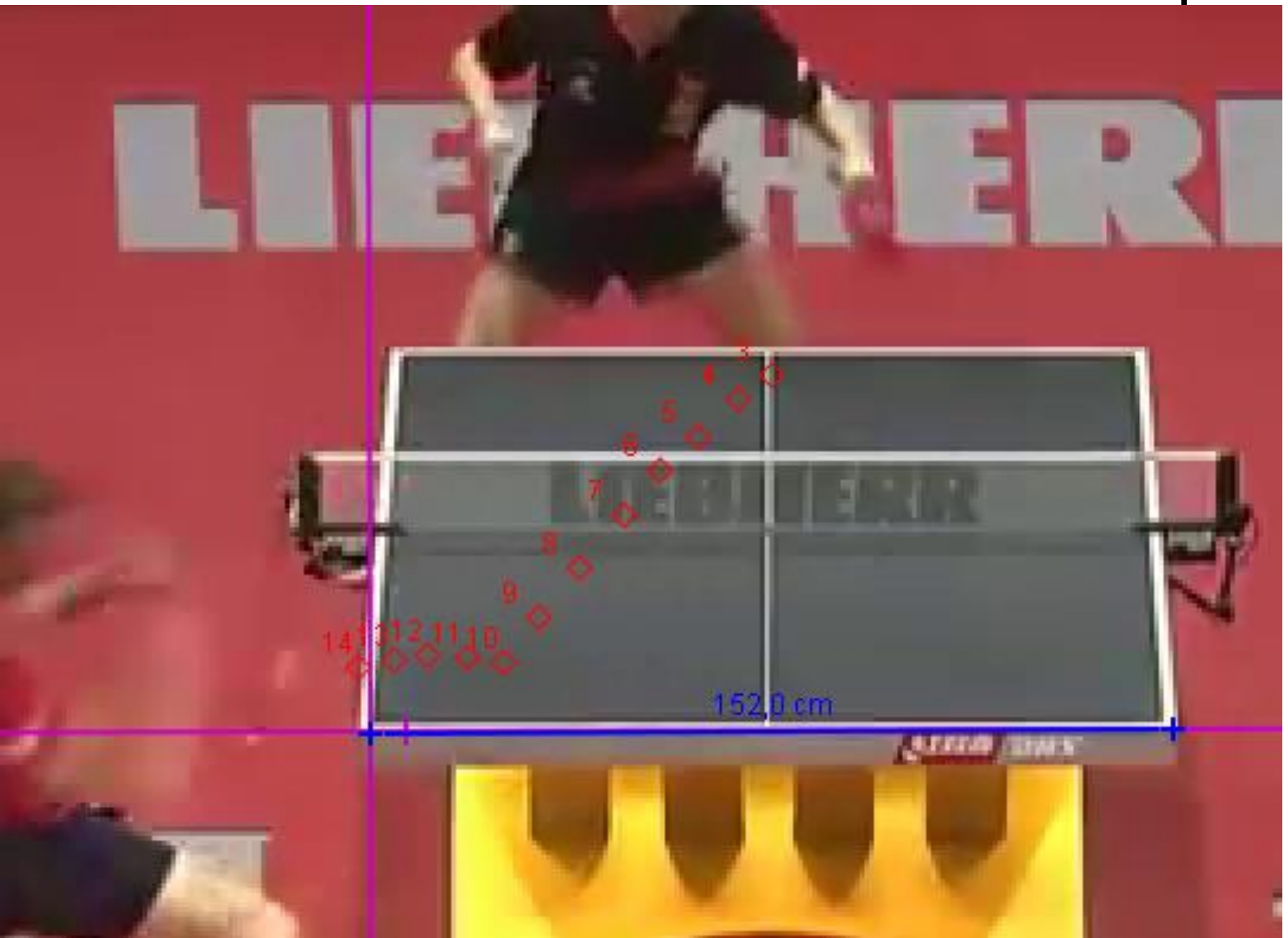
Trajectoire pointée et simulée par la théorie dans le plan de la caméra



Trajectoire pointée dans le cadre du coup 1

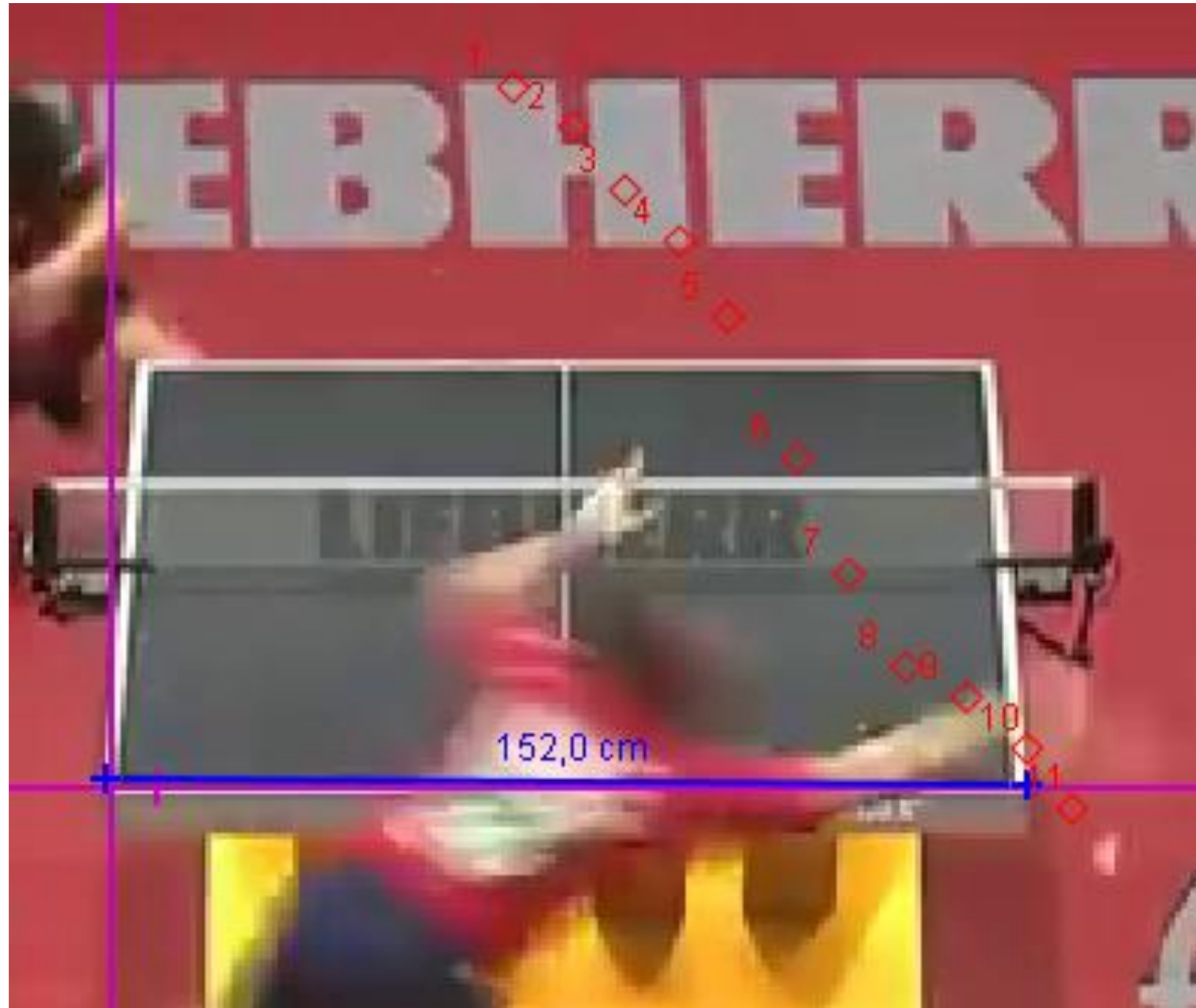
Coup 1





# Coup 3

Trajectoire pointée dans le cadre du coup 3





# Service

Trajectoire pointée dans le cadre d'un service

