Ch. 4 - Tir simple

Considérons le pb traité à la fin du Ch. 3 :

$$\begin{cases} \displaystyle\int_0^{1} u^2(t)\,dt \longrightarrow \min \\ \dot{x}(t) = -x(t) + u(t),\ t \in [0,1] \text{ (p.p.)} \\ x(0) = -1,\quad x(1) = 0 \end{cases}$$

$\longleftarrow$ t_f fixé

On a vu que $p^0 \neq 0$ (ie $p^0 < 0$: cas normal) de sorte qu'on peut prendre $p^0 = -1$ et considérer $H(x,p,u) = -u^2 + p(-x+u)$;

on a d'une part le système adjoint

$$\dot{p}(t) = p(t)$$

et la condition de maximisation d'autre part : $u(t) = p(t)/2$.

Donc : $\begin{cases} \dot{x}(t) = \nabla_p H(x(t), p(t), u(t)) = -x(t) + u(t) \\ \qquad = -x(t) + p(t)/2 = \nabla_p h(x(t), p(t)) \\ \dot{p}(t) = -\nabla_x H(\text{——}\,''\,\text{——}) = p(t) = -\nabla_x h(x(t), p(t)) \end{cases}$

avec $h(x,p) := H(x, p, p/2) \quad u(x,p) = p/2$ (cf. PMP)

$$= -(p/2)^2 + p(-x + p/2)$$

$$= p^2/4 - px$$

Rq : le caractère hamiltonien (ie la forme $\begin{cases} \dot{x} = \nabla_p h(x,p) \\ \dot{p} = -\nabla_x h(x,p) \end{cases}$ sans paramétrisation par u !) provient de la condition de maximisation :

0 car $u(x,p)$ maximiseur $\Longrightarrow \dfrac{\partial H}{\partial u}(x,p,u(x,p)) = 0$

$$\frac{\partial h}{\partial x}(x,p) = \frac{\partial H}{\partial x}(x,p,u(x,p)) + \overbrace{\frac{\partial H}{\partial u}(x,p,u(x,p))}^{0} \cdot \frac{\partial u}{\partial x}(x,p)$$

$$h(x,p) = H(x,p,u(x,p))$$

(idem pour $\dfrac{\partial h}{\partial p}$)

$\longrightarrow$ On est donc ramené à résoudre le $\overbrace{\text{pb aux deux bouts}}^{\text{TPBVP (=Two Point Boundary Value P.)}}$ suivant :

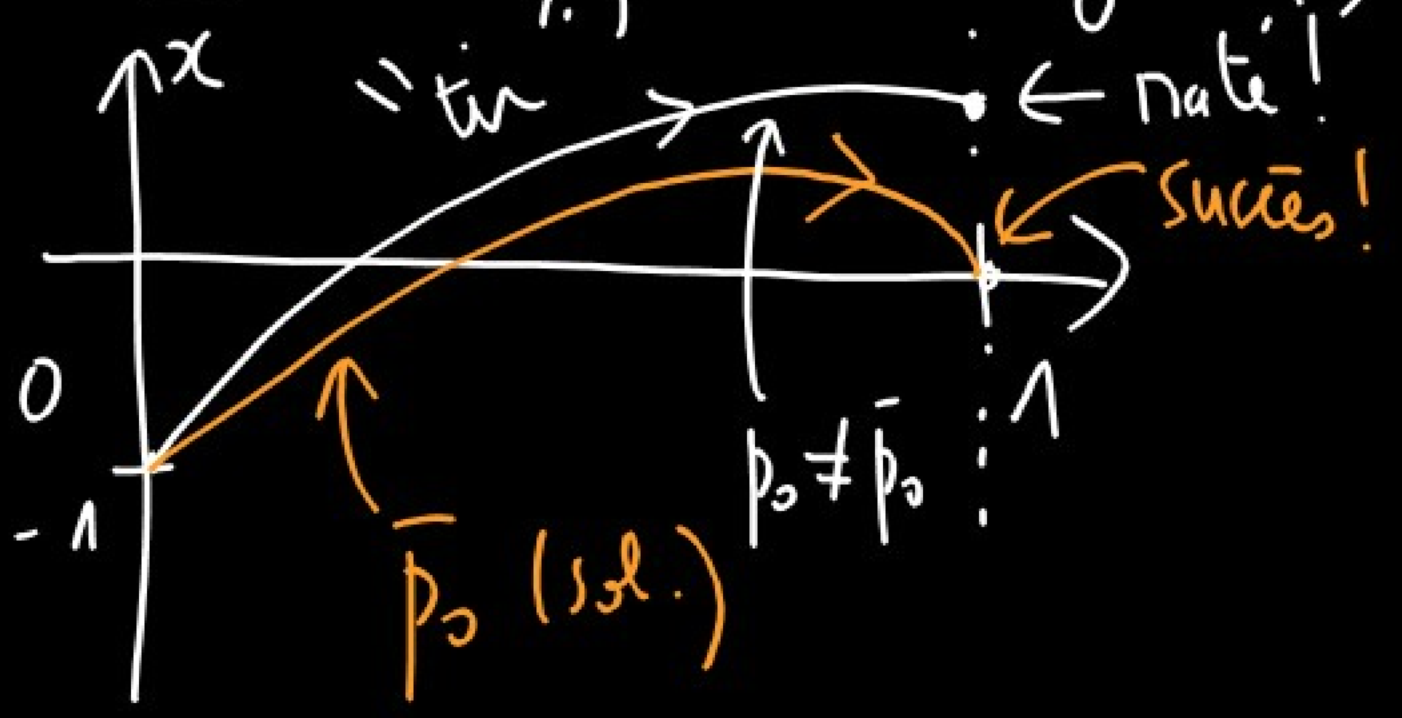
$$\begin{cases} \dot{x}(t) = D_p h(x(t), p(t)), \quad t \in [0,1] \\ \dot{p}(t) = -D_x h(x(t), p(t)) \\ x(0) = -1, \quad x(1) = 0 \end{cases}$$

Ce problème est équivalent à déterminer la condition initiale (ou finale : on pourrait travailler de manière rétrograde...) manquante, à savoir $p(0) =: p_0 \in \mathbb{R}$, tq

$t_f = 1$

$x(1, p_0) = 0$ où $x(t, p_0), p(t, p_0)$ est la sol. (cf. Cauchy-Lip.) du pb

$$\begin{cases} \dot{x}(t) = D_p h(x(t), p(t)) \\ \dot{p}(t) = -D_x h(\text{—}\,''\,\text{—}) \end{cases} \quad \text{et} \quad x(0) = -1, \; p(0) = p_0$$



On a donc à résoudre une équation à une inconnue :

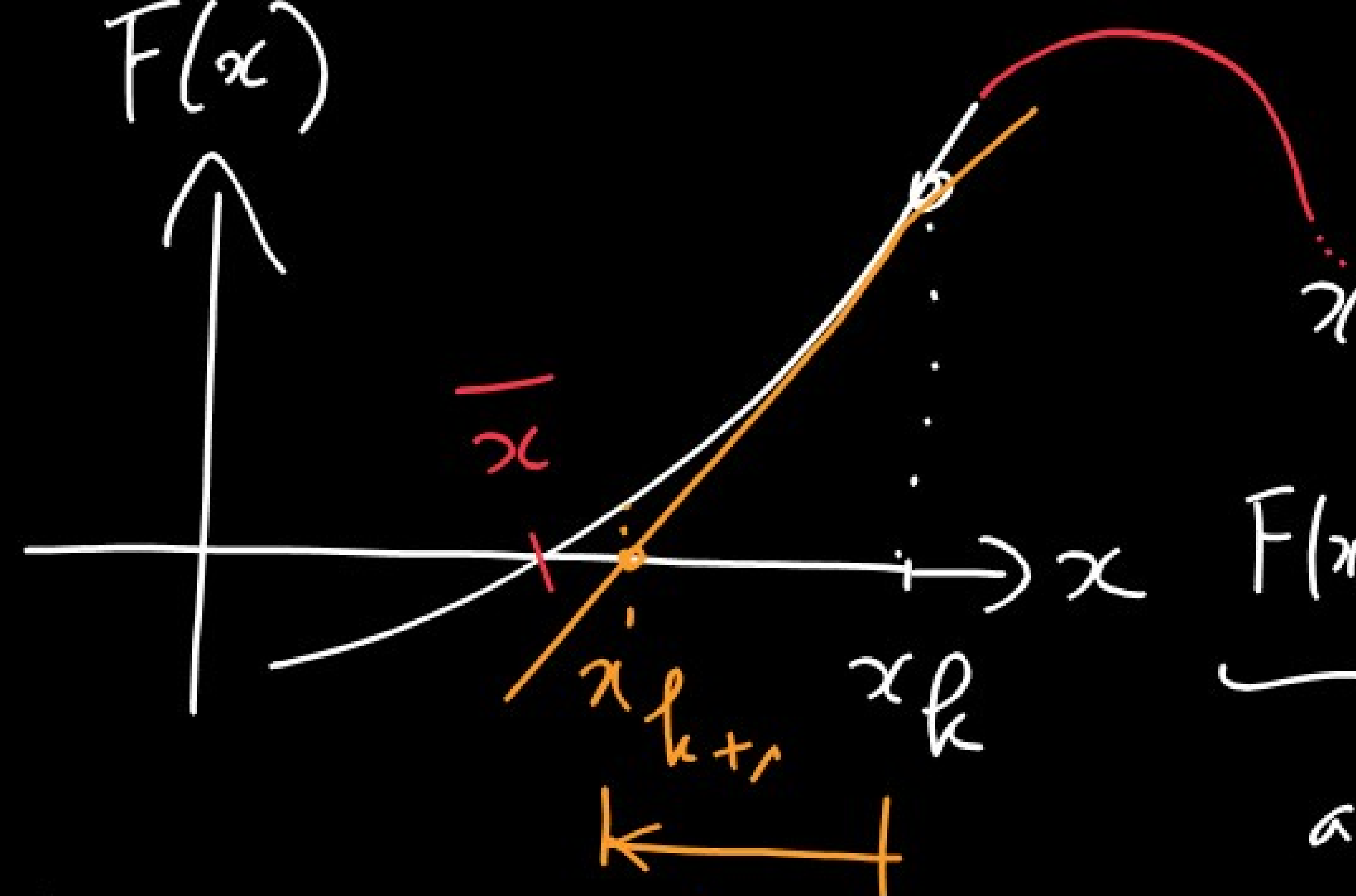
$$S(p_0) = 0$$

avec $S(p_0) := x(1, p_0)$, fonction que l'on ne connaît qu'implicitement (comme sol. d'une EDO de dim 2 en (x, p)).

En pratique :

- on évalue la fonction de tir ("shooting function") à l'aide d'un intégrateur numérique / solveur EDO (Runge-Kutta)
- on en cherche un zéro par une méthode de type Newton

Rq : i)



$x_{k+1} = x_k + d_k$ avec d_k

$\underbrace{F(x_k) + F'(x_k).d_k}_{\text{approx. affine de } F \text{ en } x_k} = 0$, ie $\boxed{F'(x_k).d_k = -F(x_k)}$

$\longrightarrow$ Factorisation (LU, QR...)

Prop : si $F : \mathbb{R}^n \to \mathbb{R}^n$ est de classe $\mathcal{C}^2$, si $F(\bar{x}) = 0$ avec $F'(\bar{x}) \in GL(n, \mathbb{R})$, alors il existe $\varepsilon > 0$ tq, $\forall x_0 \in B_f(\bar{x}, \varepsilon)$, la suite des itérés de Newton (initialisée par x_0) CV quadratiquement :

$$\|\bar{x} - x_{k+1}\| \leq C \|\bar{x} - x_k\|^2$$

choix de l'initial guess (x_0) critique et délicat...

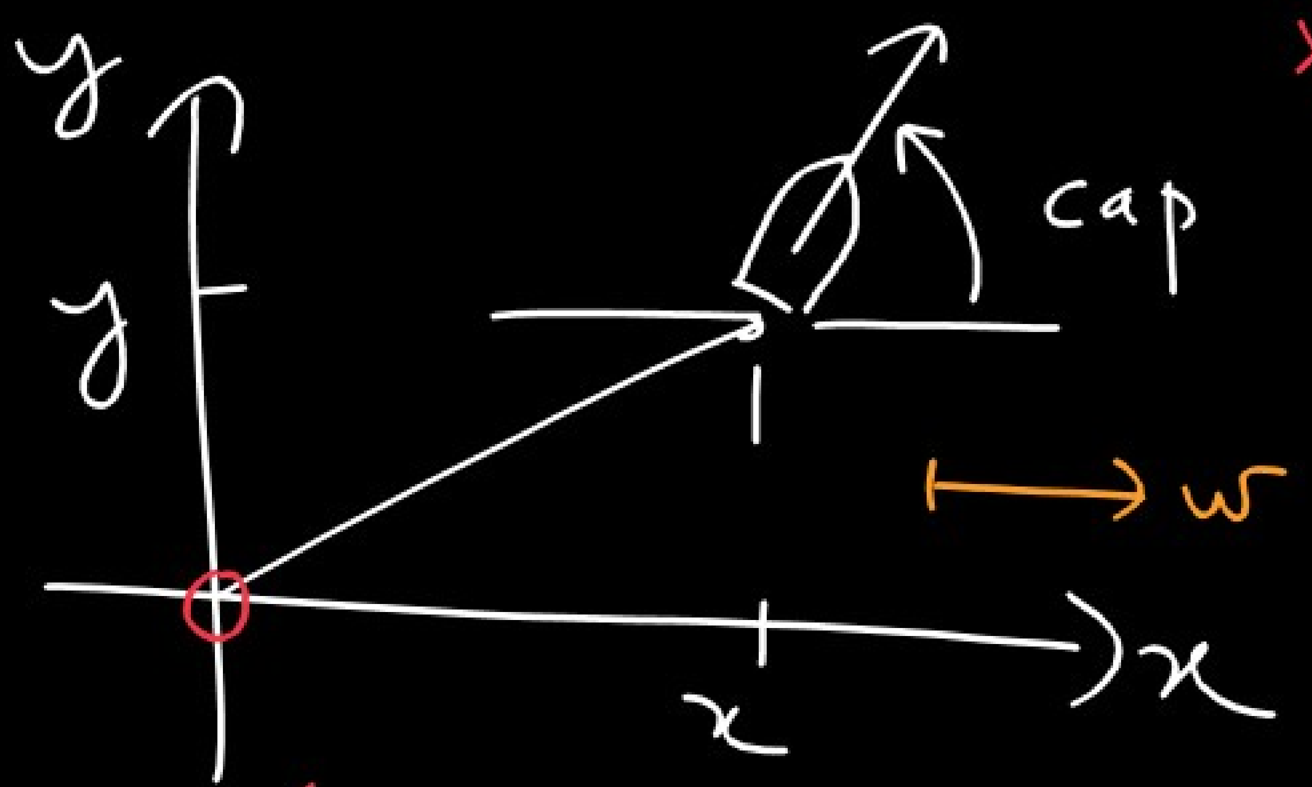
Pour "globaliser" (= rendre moins locale cette CV) on peut notamment considérer le pb des moindres carrés nonlinéaires : $\frac{1}{2}\|F(x)\|^2 \longrightarrow \min$.

ii) ici, le calcul de $S'(p_0)$, dérivée $\dfrac{\partial x(t=1,p_0)}{\partial p_0}$ de la fonction de tir peut (doit...) se faire par différentiation automatique (AD / differentiable programming)

$\longrightarrow$ génération du code la dérivée de S (ici en utilisant le th. des fonctions implicites) à partir du code de S.

iii) idem en dimension n (avec, par ex., $p(0) =: p_0 \in \mathbb{R}^n$ ...)

Ch. 5 – Model Predictive Control (MPC)



$x_f, y_f \searrow \theta_f$

$$\begin{cases} \dot x(t) = \overset{w_x}{\cancel{w}} + \overset{1}{\cancel{V}}\cdot\cos(\theta(t)) \\ \dot y(t) = w_y \quad \underset{1}{\cancel{V}}\cdot\sin(\theta(t)) \\ \dot\theta(t) = u(t), \quad |u(t)| \le 1 \end{cases}$$

$\|\cdot\| \le 1$

→ en réalité, le courant est non seulement (w_x, w_y) avec $w_y \neq 0$, mais aussi avec w_x, w_y qui dépendent de la position (voire du temps...)

→ comment prendre en compte les mesures de l'état et des paramètres pour atteindre effectivement la cible ?

Algo MPC :

- on résout le pb de $X_0 = (x_0, y_0, \theta_0)$ à $X_f = (x_f, y_f, \theta_f)$
  $\longrightarrow t_f$ (min) et $u : [0, t_f] \longrightarrow \mathbb{R}$
- on injecte ce contrôle dans le "vrai" système (= ici intégration d'un système avec modèle de courant non constant / Simulation) non pas sur tout $[0, t_f]$ (risque de grande déviation !) mais uniquement sur une fraction du temps final ($\Delta t \ll t_f$)
- on mesure $X(\underbrace{t_0 + \Delta t}_{\ll t_f})$ et $w = (w_x, w_y)$ en ce pt (déviation inévitable : aucun modèle n'est pas parfait !) ; on pose $X_1 = X(t_0 + \Delta t)$

- on résout à nouveau le pb entre X_1 et X_f

  ↳ on en tire un nouveau u, un nouveau t_f qu'on applique sur $[\underset{\substack{\uparrow \\ t_0+\Delta t}}{t_1}, t_1+\Delta t]$

- on recommence

- test d'arrêt : $\| \underset{\substack{\uparrow \\ n^{\text{ème}} \text{ itéré}}}{X_n} - X_f \| \leq \varepsilon \quad (+ \, n \leq N_{max})$