

3. Génération de trajectoires en robotique

Ludovic Hofer

**Master 2,
Université de Bordeaux**

Automne 2021 - version 1.1

Références

- [Boo01] Carl de Boor. *A Practical Guide to Splines - Revised Edition*. 2001. ISBN: 9780387953663.
- [Cra04] John J. Craig. *Introduction to Modern Robotics: Mechanics, Planning, and Control*. 2004. ISBN: 978-0201543612.
- [Cro] Jean-Pierre Croisille. *2-INTERPOLATION SPLINE*. URL: <http://www.math.univ-metz.fr/~croisil/M1-0809/2.pdf>.
- [Den] Timo Denk. *Cubic Spline Interpolation – Timo Denk’s Blog*. URL: <https://timodenk.com/blog/cubic-spline-interpolation/> (visited on 11/09/2020).
- [KDN03] W Khalil, E Dombre, and ML Nagurka. “Modeling, Identification and Control of Robots”. In: *Applied Mechanics Reviews* (2003). ISSN: 0003-6900. DOI: 10.1115/1.1566397.
- [Nik01] Saeed B. Niku. *Introduction to Robotics: Analysis, Control, Applications*. Prentice Hall New Jersey, 2001. ISBN: 9781119527626.

Sources et avertissements

Sources:

- Sources générales
 - [Cra04] Chapitre 7
 - [KDN03] Chapitre 13
- Sources supplémentaires pour les splines
 - [Den; Cro; Boo01]

Avertissements

- Ce cours présente seulement une partie "basique" des générations de trajectoires, il ne décrit aucune méthode avancée.
- Dans certains cas, il existe de nombreuses approches, ce cours présente un peu la diversité des méthodes, mais en approfondit qu'un nombre limité.

Problématique

Objectifs

- Simplifier l'interface utilisateur
- Minimiser la durée d'exécution
- Éviter les discontinuités (position, vitesse, accélération, ...)

Contraintes

- Pour obtenir des mouvements réalisables, prendre en compte les **limites physiques**
 - Les couples applicables sont limités
 - Il y a des vitesses maximales
- Respecter des contraintes fortes en terme de **temps de calcul**
 - Fréquence habituelle: 60Hz à 2000Hz [Cra04]

Génération de trajectoires

Exemples de cibles utilisateur

- Début et fin de la trajectoire: $x_{\text{src}}, x_{\text{end}}$
- Points de passages: $((t_0, x_0), (t_1, x_1), \dots, (t_n, x_n))$

Résultat attendu:

Une trajectoire avec une expression simple à calculer pour:

- La durée T
- Dans l'intervalle de temps $[0, T]$:

$x(t)$ La position

$\dot{x}(t)$ La vitesse

$\ddot{x}(t)$ L'accélération

$\dddot{x}(t)$ Le jerk

Trajectoires et continuités

Continuité au degré n

On note $C^n[t_a, t_b]$ l'ensemble des fonctions continues et dérivable à l'ordre n sur l'intervalle $[t_a, t_b]$.

Signification

- Pour une trajectoire \mathcal{T} :
 - $\mathcal{T} \notin C^0[0, T] \implies$ vitesse infinie
 - $\mathcal{T} \notin C^1[0, T] \implies$ accélération infinie
 - $\mathcal{T} \notin C^2[0, T] \implies$ jerk infini, variation de force instantanée

Les trajectoires sont définies dans un espace choisi

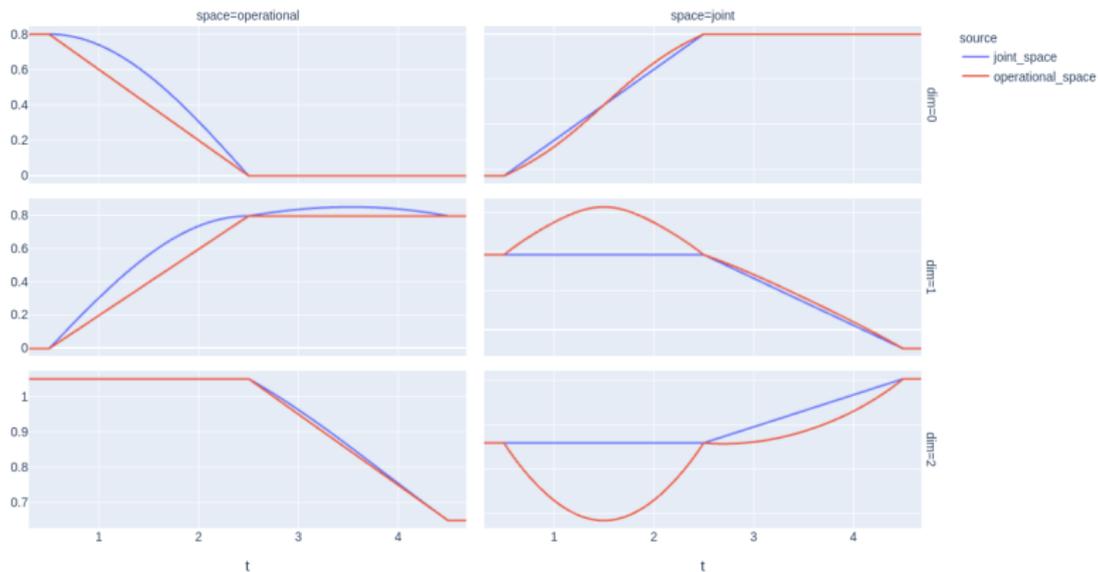
Trajectoire dans l'espace articulaire

- La trajectoire d'une articulation est indépendante des autres
- Il est facile d'établir des limites sur la vitesse et l'accélération de chaque articulation
- Quand la trajectoire de l'outil entre les points de passage importe peu.

Trajectoire dans l'espace opérationnel

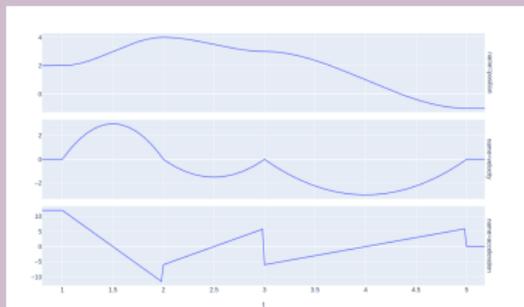
- Quand la trajectoire de l'outil entre les points de passage est contrainte.
- Plusieurs problèmes possibles: (voir [Cra04])
 - La trajectoire passe par des points non-atteignables
 - Vitesse infinie des articulations nécessaire près des singularités
 - Le nombre de solutions au MGI peut varier

Espace des trajectoires \neq Espace des points de passage

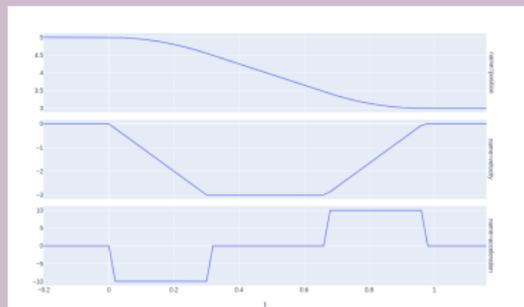


Comment décrire ces trajectoires? Quelques exemples:

Splines cubiques



Lois de mouvement trapézoïdales



Présentation

- Une spline est spécifiée par des **points de passages**
 - points de passage (via points) \neq points de contrôle
 - Dans certains cas, des informations supplémentaires sont transmises à travers les points de contrôle (ex. dérivée à chaque point)
- La trajectoire est obtenue à partir des points de passage et d'une **méthode d'interpolation**
 - Définie par le degré des polynômes utilisés et parfois des contraintes supplémentaires pour résoudre des ambiguïtés
 - Le choix de la méthode d'interpolation définit à quel degré la trajectoire est continue.
- La trajectoire entre les points de passage est définie localement par des **polynômes**
- Tout polynôme est C^∞ , les problèmes de continuité auront donc lieu sur les points de passage.

Formalisme

Points de passage

$$P = ((t_0, x_0), (t_1, x_1), \dots, (t_n, x_n))$$

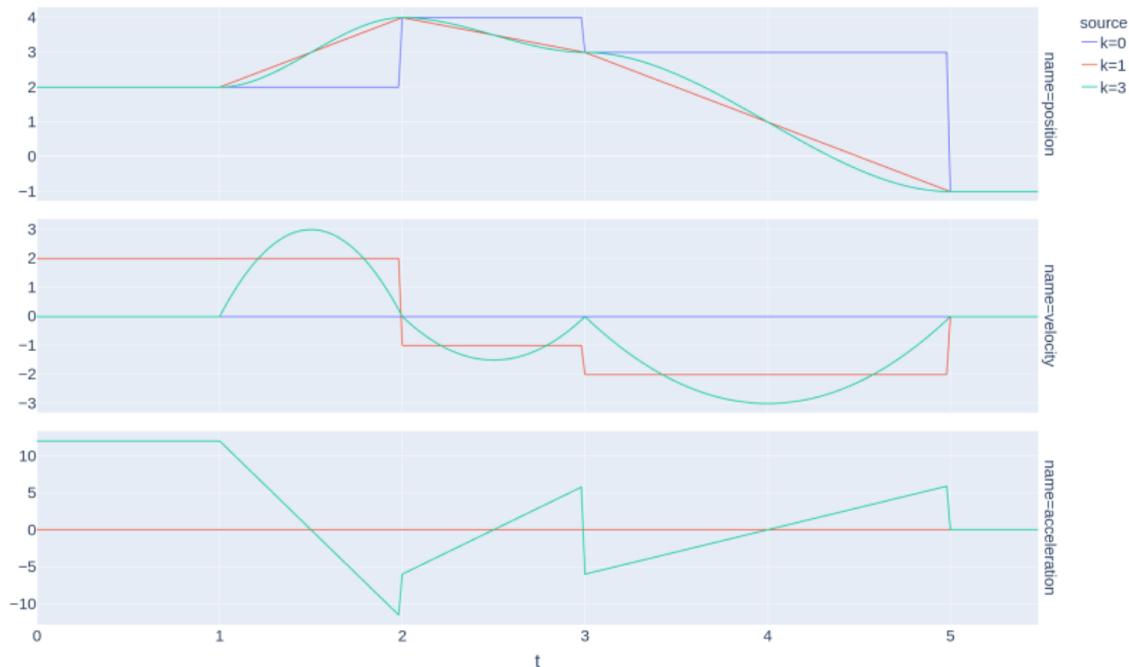
Résultat

Une fonction $S : [0, T] \rightarrow \mathbb{R}$ est une spline de degré k si:

- $0 = t_0 < t_1 < \dots < t_n = T$

- $$S(t) = \begin{cases} S_1(t) & t_0 \leq t < t_1 \\ \vdots & \vdots \\ S_n(t) & t_{n-1} \leq t \leq t_n \end{cases}$$

- Pour tout i de 1 à n , S_i est un polynôme de degré k .

Exemple $k = 0$, $k = 1$, $k = 3$ 

Splines constantes

Points de passage

- Classique: $P = ((t_0, x_0), (t_1, x_1), \dots, (t_n, x_n))$

Méthode d'interpolation

La dernière valeur spécifiée est utilisée jusqu'au prochain point de passage:

$$S_i(t) = x_i$$

Remarques

Cet exemple est avant tout pédagogique, ce n'est pas une manière convenable de définir des trajectoires.

Splines linéaires

Points de passage

- Classique: $P = ((t_0, x_0), (t_1, x_1), \dots, (t_n, x_n))$

Méthode d'interpolation

- Pour assurer que la trajectoire $\mathcal{T} \in C^0[t_0, t_n]$ il faut assurer:
 - $S_i(t_i) = x_i$
 - $S_i(t_{i+1}) = x_{i+1}$
- Ceci implique que $S_i(t) = (t - t_i) \frac{x_{i+1} - x_i}{t_{i+1} - t_i} + x_i$

Remarques

Simple à définir et à utiliser, les splines linéaires permettent d'assurer une continuité des ordres, mais demandent une accélération infinie.

Splines cubiques: méthodes locales vs méthodes globales

Commun

- Il y a 4 variables libres pour chaque polynôme donc $4n$ variables

Méthodes locales

- Le couple de point $((x_i, y_i), (x_{i+1}, y_{i+1}))$ définit S_i
- Résolution de n systèmes d'équations:
 - 4 équations, 4 inconnues

Méthodes globales

- Les coefficients de chaque polynôme S_i dépendent de l'ensemble des points
- Résolution d'un seul système d'équations
 - $4n$ équations, $4n$ inconnues

Splines cubiques locales

Localement: 2 Contraintes, 4 degrés de liberté

- Contraintes pour $C^0[t_0, t_n]$
 - $S_i(t_i) = x_i$
 - $S_i(t_{i+1}) = x_{i+1}$
- 4 variables libre par polynôme: $S_i(t) = a_i t^3 + b_i t^2 + c_i t + d_i$
- Il reste deux degrés de liberté

Exemples de contraintes supplémentaires

- Imposer $\dot{S}_i(t_i) = 0$ et $\dot{S}_i(t_{i+1}) = 0$
- Choisir manuellement $\dot{S}_i(t_i)$
- Étendre le voisinage: (Voir [Cro], Méthode 1)
 - $S_i(t_{i-1}) = x_{i-1}$
 - $S_i(t_{i+2}) = x_{i+2}$

Exemple: Splines cubiques avec dérivées nulles

Points de passage

- Classique: $P = ((t_0, x_0), (t_1, x_1), \dots, (t_n, x_n))$

Méthode d'interpolation pour S_i

$$\begin{pmatrix} t_i^3 & t_i^2 & t_i & 1 \\ t_{i+1}^3 & t_{i+1}^2 & t_{i+1} & 1 \\ 3t_i^2 & 2t_i & 1 & 0 \\ 3t_{i+1}^2 & 2t_{i+1} & 1 & 0 \end{pmatrix} \begin{pmatrix} a_i \\ b_i \\ c_i \\ d_i \end{pmatrix} = \begin{pmatrix} x_i \\ x_{i+1} \\ 0 \\ 0 \end{pmatrix} \quad (1)$$

Remarques

La trajectoire obtenue est $C^1[t_0, t_n]$

Exemple pratique: Splines cubiques avec dérivées nulles

Input:

$$P = ((0, 2), (1, 3), (4, -2))$$

Déterminer S_0 et S_1

$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{pmatrix} \begin{pmatrix} a_0 \\ b_0 \\ c_0 \\ d_0 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 0 \\ 0 \end{pmatrix} \quad (2)$$

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 64 & 16 & 4 & 1 \\ 3 & 2 & 1 & 0 \\ 48 & 8 & 1 & 0 \end{pmatrix} \begin{pmatrix} a_1 \\ b_1 \\ c_1 \\ d_1 \end{pmatrix} = \begin{pmatrix} 3 \\ -2 \\ 0 \\ 0 \end{pmatrix} \quad (3)$$

Résultats et vérifications

- Polynômes obtenus:
 $S_0(t) = -2t^3 + 3t^2 + 2$
 $S_1(t) = 0.4t^3 - 2.8t^2 + 4.4t + 1.0$
- Vérification dérivée:
 $\dot{S}_0(1) = -3 \cdot 2 + 2 \cdot 3 = 0$
 $\dot{S}_1(1) = 3 \cdot 0.4 - 2 \cdot 2.8 + 4.4 = 0$
- Dérivée seconde discontinue:
 $\ddot{S}_0(1) = -6 \cdot 2 + 2 \cdot 3 = -6$
 $\ddot{S}_1(1) = 6 \cdot 0.4 - 2 \cdot 2.8 = -3.2$

Splines cubiques globales

Globalement: $4n - 2$ contraintes, $4n$ variables

- Contraintes de position: **$2n$ équations**
 - Pour $i \in \llbracket 0, n - 1 \rrbracket$, $S_i(t_i) = x_i$
 - Pour $i \in \llbracket 0, n - 1 \rrbracket$, $S_i(t_{i+1}) = x_{i+1}$
- Contraintes de continuité des dérivées: **$2n - 2$ équations**
 - Pour $i \in \llbracket 1, n - 1 \rrbracket$, $S_{i-1}^{\dot{}}(t_i) = \dot{S}_i(t_i)$
 - Pour $i \in \llbracket 1, n - 1 \rrbracket$, $S_{i-1}^{\ddot{}}(t_i) = \ddot{S}_i(t_i)$

Exemples de contraintes supplémentaires

- Splines naturelles: $\ddot{S}_0(t_0) = 0$ et $S_{n-1}^{\ddot{}}(t_n) = 0$
- Splines périodiques: $\dot{S}_0(t_0) = \dot{S}_{n-1}(t_n)$ et $\ddot{S}_0(t_0) = \ddot{S}_{n-1}(t_n)$

Exemple: splines cubiques naturelles

Méthode d'interpolation pour S_i

$$\begin{pmatrix}
 t_0^3 & t_0^2 & t_0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\
 t_1^3 & t_1^2 & t_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\
 0 & 0 & 0 & 0 & t_1^3 & t_1^2 & t_1 & 1 & 0 & 0 & 0 & 0 & \dots \\
 0 & 0 & 0 & 0 & t_2^3 & t_2^2 & t_2 & 1 & 0 & 0 & 0 & 0 & \dots \\
 \vdots & \dots \\
 \vdots & \dots \\
 3t_1^2 & 2t_1 & 1 & 0 & -3t_1^2 & -2t_1 & -1 & 0 & 0 & 0 & 0 & 0 & \dots \\
 6t_1 & 2 & 0 & 0 & -6t_1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\
 0 & 0 & 0 & 0 & 3t_2^2 & 2t_2 & 1 & 0 & -3t_2^2 & -2t_2 & -1 & 0 & \dots \\
 0 & 0 & 0 & 0 & 6t_2 & 2 & 0 & 0 & -6t_2 & -2 & 0 & 0 & \dots \\
 \vdots & \dots \\
 \vdots & \dots \\
 6t_0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots
 \end{pmatrix}
 \begin{pmatrix}
 a_0 \\
 b_0 \\
 c_0 \\
 d_0 \\
 a_1 \\
 b_1 \\
 c_1 \\
 d_1 \\
 \vdots
 \end{pmatrix}
 =
 \begin{pmatrix}
 x_0 \\
 x_1 \\
 x_2 \\
 \vdots \\
 0 \\
 0 \\
 0 \\
 0 \\
 0 \\
 \vdots \\
 0 \\
 0
 \end{pmatrix} \quad (4)$$

Remarques

La trajectoire obtenue est $C^2]$ $t_0, t_n[$

Quelques commentaires

Interface utilisateur

- Ajouter un point est facile mais anticiper l'impact sur la trajectoire devient difficile dès que le degré est supérieur à 1.

Réalisabilité

Si la continuité des trajectoires est intéressante, il n'y a aucune garantie qu'elles soient "réalisables".

Degré > 3 : Beaucoup de méthodes d'interpolation différentes

En théorie, cette diversité offre la possibilité d'adapter ce choix en fonction du contexte. Mais, en pratique, il est souvent difficile de décider de la meilleure méthode.

Présentation

Objectifs:

Générer une trajectoire \mathcal{T} :

- D'une source $\mathcal{T}(0) = x_{\text{src}}$
- À une cible: $\mathcal{T}(T) = x_{\text{end}}$
- Dont la durée est minimale (optimalité)
- Qui respecte un ensemble de contraintes (réalisabilité)

Exemples de contraintes:

- $\forall t \in [0, T], |\dot{\mathcal{T}}(t)| \leq v_{\text{max}}$: Limite de vitesse
- $\forall t \in [0, T], |\ddot{\mathcal{T}}(t)| \leq \alpha_{\text{max}}$: Limite d'accélération
- $\forall t \in [0, T], |\dddot{\mathcal{T}}(t)| \leq j_{\text{max}}$: Limite de jerk

Loi trapézoïdale en vitesse: Présentation

Contraintes spécifiques

- $\forall t \in [0, T], |\dot{T}(t)| \leq v_{\max}$: Limite de vitesse
- $\forall t \in [0, T], |\ddot{T}(t)| \leq \alpha_{\max}$: Limite d'accélération
- $\dot{T}(0) = 0$ et $\dot{T}(T) = 0$: Les vitesses initiales et finales sont nulles

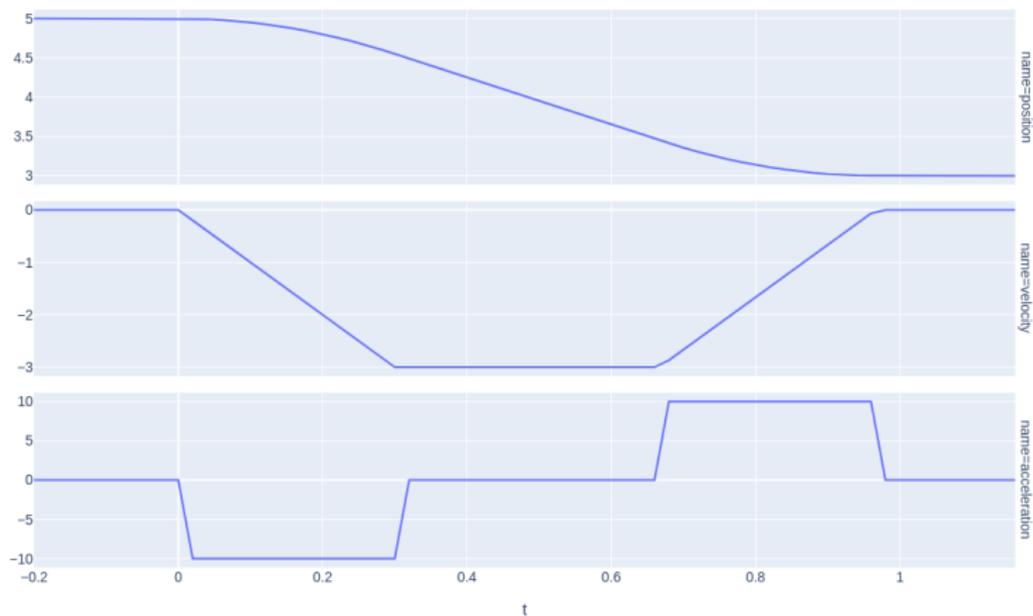
Intuition générale

Pour minimiser le temps, accélérer à fond jusqu'à atteindre la vitesse maximale. Puis, décélérer à fond au dernier moment.

Précautions à prendre

- La distance entre x_{src} et x_{end} peut être insuffisante pour atteindre v_{\max}
- La distance entre x_{src} et x_{end} peut être négative

Loi trapézoïdale en vitesse: représentation graphique



Loi trapézoïdale en vitesse: calculs I

Est-ce qu'on a le temps d'atteindre la vitesse maximale?

Dépend de la valeur absolue de la distance $D = x_{\text{end}} - x_{\text{src}}$:

- Accélération durant t , distance parcourue: $d_{\text{acc}} = \frac{1}{2}\alpha_{\text{max}}t^2$
- Pareil pour la décélération, donc $d_{\text{acc}} + d_{\text{dec}} = \alpha_{\text{max}}t^2$
- Pour une accélération durant t , la vitesse atteinte est: $v = \alpha_{\text{max}}t$
- Donc si v_{max} est atteinte, alors $t = \frac{v_{\text{max}}}{\alpha_{\text{max}}}$
- Ce qui signifie que $|D| \geq \alpha_{\text{max}} \frac{v_{\text{max}}^2}{\alpha_{\text{max}}^2} = \frac{v_{\text{max}}^2}{\alpha_{\text{max}}}$

Loi trapézoïdale en vitesse: calculs II

Durée de la phase d'accélération? τ_α ?

Si on a le temps d'atteindre la vitesse maximale:

$$\tau_\alpha = \frac{v_{\max}}{\alpha_{\max}}$$

Sinon, on peut utiliser $|D| = d_{\text{acc}} + d_{\text{dec}} = \alpha_{\max} \tau_\alpha^2$:

$$\tau_\alpha = \sqrt{\frac{|D|}{\alpha_{\max}}}$$

Distance parcourue pendant l'accélération? D_α

En reprenant les formules précédentes on a simplement $D_\alpha = \frac{\alpha_{\max} \tau_\alpha^2}{2}$

Loi trapézoïdale en vitesse: calculs III

Quelle est la durée de la trajectoire entière? T

- La distance parcourue à vitesse maximale est simplement: $|D| - 2D_\alpha$
- La durée totale de la trajectoire est donc: $T = 2\tau_\alpha + \frac{|D| - 2D_\alpha}{v_{\max}}$

On a donc trois phases clairement identifiées:

- 1 L'accélération: $\mathcal{T}(t) = x_{\text{src}} + \text{sign}(D) \frac{\alpha_{\max} t^2}{2}$
- 2 La phase à vitesse maximale:
 $\mathcal{T}(t) = x_{\text{src}} + \text{sign}(D)(D_\alpha + v_{\max}(t - \tau_\alpha))$
 - Seulement si $|D| \geq \frac{v_{\max}^2}{\alpha_{\max}}$
- 3 La décélération: $\mathcal{T}(t) = x_{\text{end}} - \text{sign}(D) \frac{\alpha_{\max}(T-t)^2}{2}$
 - Symétrique à l'accélération

Loi trapézoïdale en vitesse: Récapitulatif

Paramètres d'entrée

- x_{src} position initiale
- x_{end} cible
- v_{max} vitesse maximale
- α_{max} accélération maximale

Éléments importants

$$D = x_{end} - x_{src}$$

$$\tau_{\alpha} = \begin{cases} \frac{v_{max}}{\alpha_{max}} & \text{si } |D| > \frac{v_{max}^2}{\alpha_{max}} \\ \sqrt{\frac{|D|}{\alpha_{max}}} & \text{sinon} \end{cases}$$

$$D_{\alpha} = \frac{\alpha_{max} \tau_{\alpha}^2}{2}$$

$$T = 2\tau_{\alpha} + \frac{|D| - 2D_{\alpha}}{v_{max}}$$

Trajectoire finale

$$\mathcal{T}(t) = \begin{cases} x_{src} + \text{sign}(D) \frac{\alpha_{max} t^2}{2} & \text{si } t \leq \tau_{\alpha} \\ x_{end} - \text{sign}(D) \frac{\alpha_{max} (T-t)^2}{2} & \text{si } t > 1 - \tau_{\alpha} \\ x_{src} + \text{sign}(D) (D_{\alpha} + v_{max}(t - \tau_{\alpha})) & \text{sinon} \end{cases} \quad (5)$$

Synchronisation de trajectoires multi-dimensionnelles

Observation

- Les trajectoires en robotique concernent généralement plusieurs dimensions, on a donc une trajectoire par dimension: $\mathcal{T}_1, \mathcal{T}_2, \dots$
- Planifier les trajectoires indépendamment engendre amène à ce que chaque trajectoire \mathcal{T}_i ait sa propre durée maximale T_i

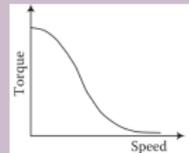
Synchronisation

- Il est généralement préférable de les synchroniser pour minimiser les accélérations et vitesses demandées aux différents joints. Ce qui permet:
 - De réduire la consommation d'énergie.
 - D'obtenir plus de précision dans le suivi d'une trajectoire.
- Idéalement, on essaie aussi de synchroniser la durée d'accélération τ_α des différents joints pour avoir un nombre de phases distinctes limité.

Des trajectoires réalisables et optimales: vraiment?

Les constantes v_{\max} et α_{\max} sont des approximations

Les limites se situent plutôt au niveau du couple disponibles au niveau des moteurs, couple qui est influencé par la vitesse de rotation.



Une courbe typique du torque en fonction de la vitesse pour un moteur à pas (Source: [Nik01], Fig 9.30)

La configuration du robot n'est pas anodine

Si un bras robotique est "tendu", le couple nécessaire ne sera pas le même que s'il est replié.

Des forces s'exercent sur le robot

La gravité peut aussi bien aider que contrer un mouvement, considérer α_{\max} au niveau d'un joint est donc une approximation insatisfaisante.