

ÉTAT DE L'ART SUR LA CINÉMATIQUE INVERSE EN TÉLÉOPÉRATION

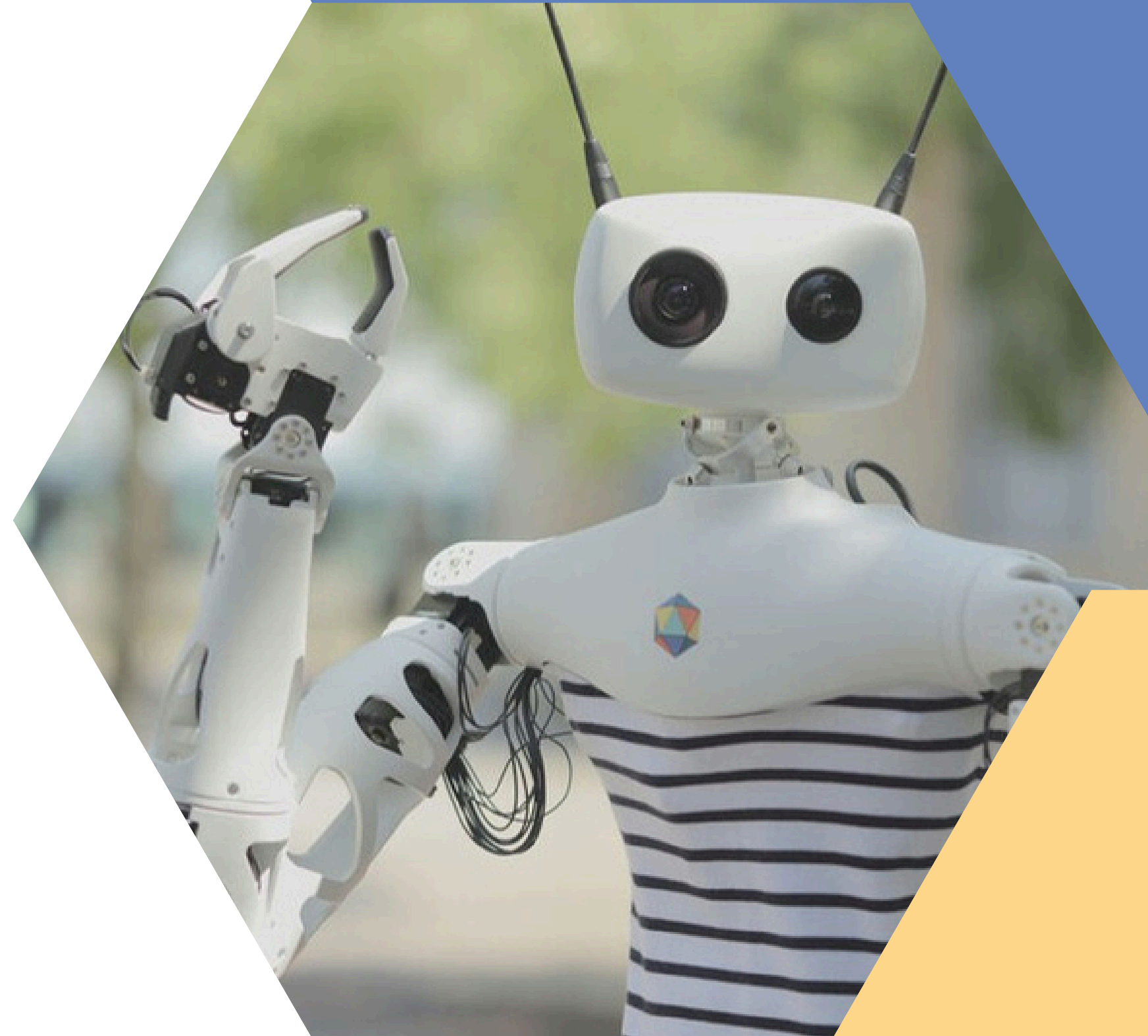
SOUTENANCE PROJET ROBOTIQUE

Présenté par Kloé Bonnet et Guillaume Lauga

13/12/2024

Encadrement :

- Rémi Fabre - Ingénieur en robotique à Pollen Robotics
- Vincent Padois - Chercheur à Inria - Auctus



Contextes

Reachy2 : robot antropomorphe, code *open source*, développé par Pollen Robotics, contrôlable par téléopération via un casque VR



Reachy2 : Robot conçu par Pollen Robotics

- Assurer la continuité de la loi de commande
 - Gérer les singularités et les limites articulaires
 - Cinématique inverse analytique utilisée, mais limites articulaires partiellement gérées.
 - Manque de robustesse et de flexibilité dans la gestion des contraintes physiques.
 - Sensibilité aux singularités entraînant des mouvements disproportionnés.
- ↪
- Étude des alternatives pour la cinématique inverse adaptée à Reachy2.
 - Propositions de solutions robustes et flexibles pour gérer les contraintes articulaires.



POLLEN
ROBOTICS

1

**Problèmes liés à la
téléopération**

2

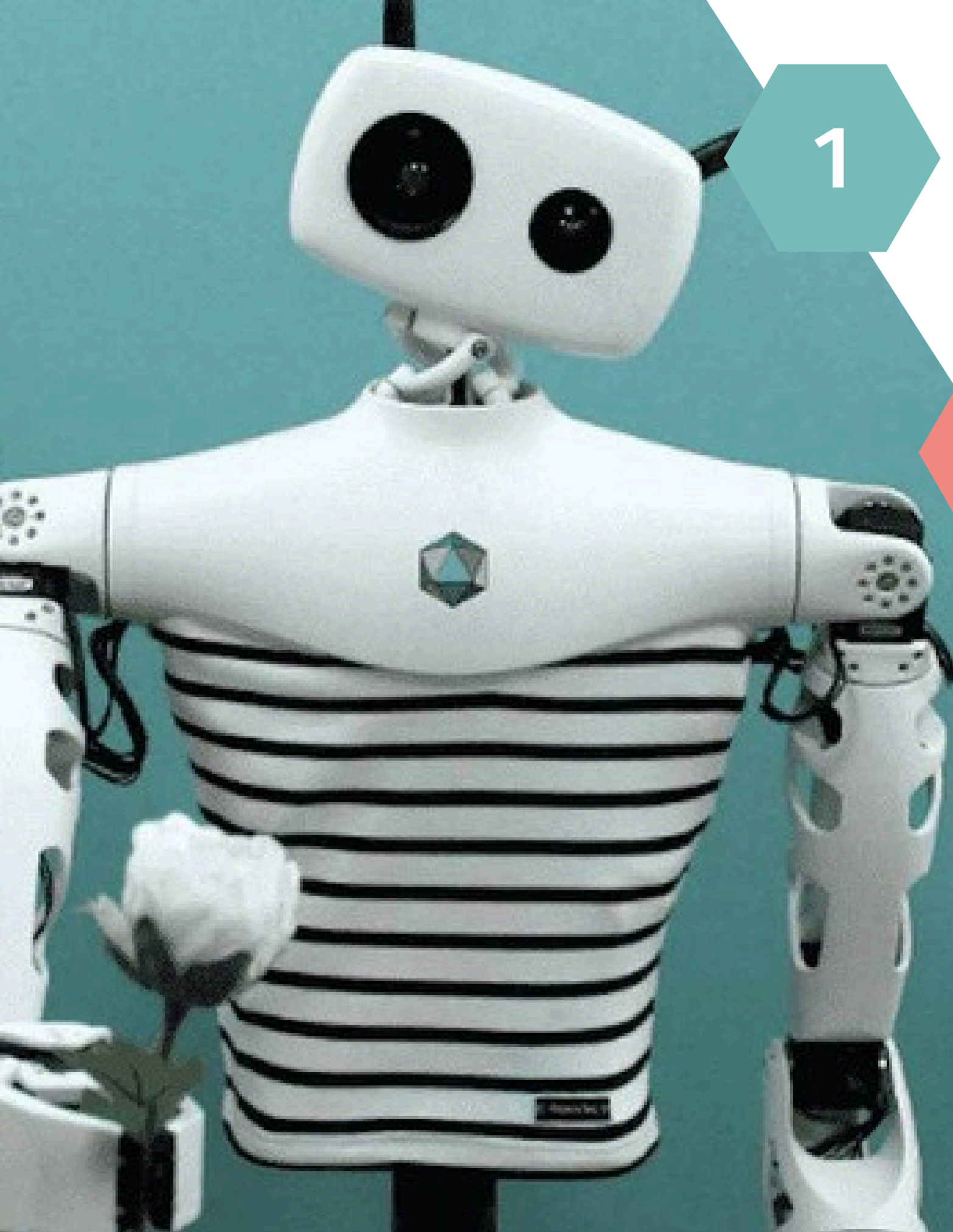
**Cinématique Inverse :
Approches et Limitations**

3

**Implémentation de contraintes
et leurs résolutions**

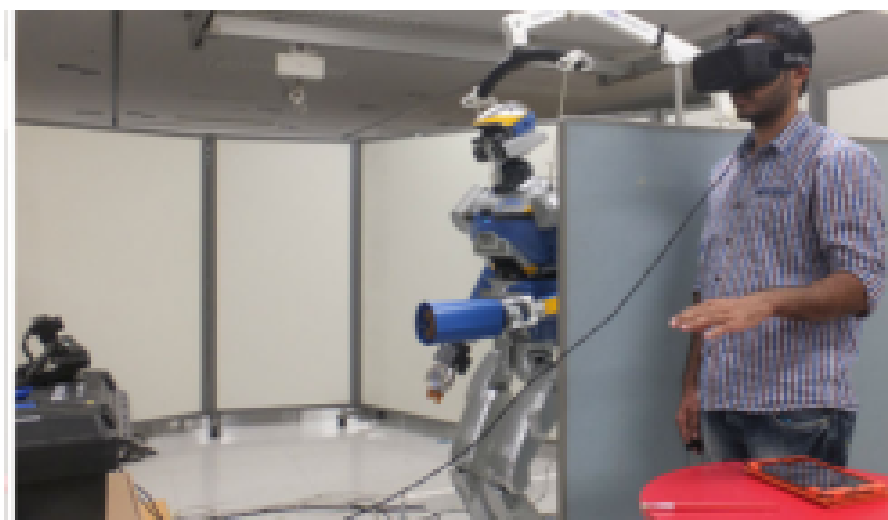
4

**Conclusion &
perspectives**





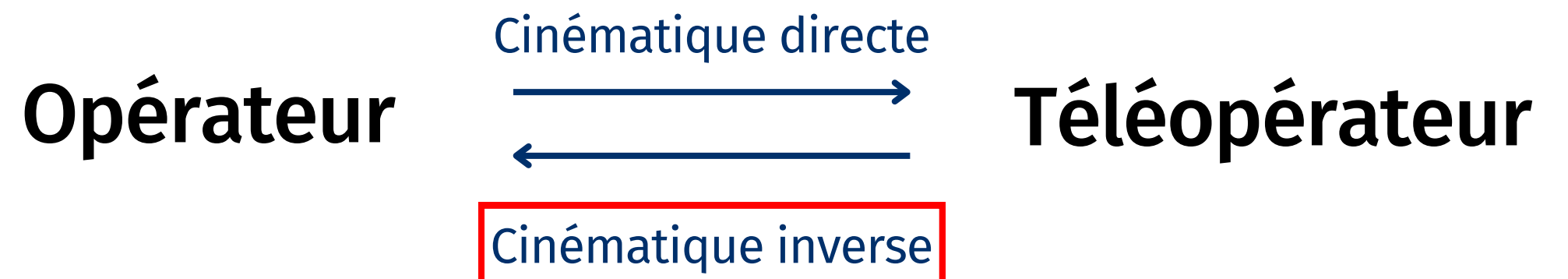
Téléopérateur et opérateur



Exemples de système téléopérés

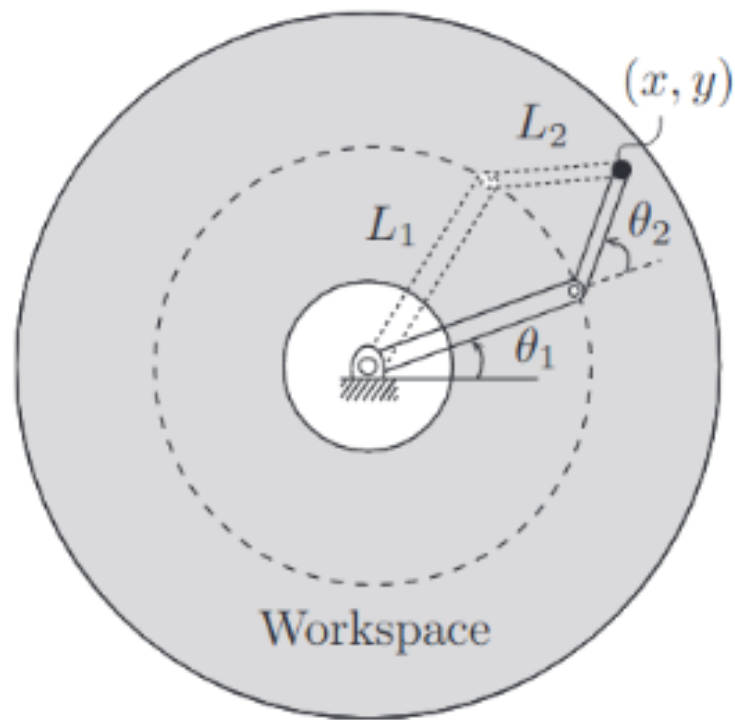
Définitions

- Téléopération : “Faire un travail à distance” ou “contrôler un véhicule ou un système externe à distance”
- Opérateur : la personne qui contrôle et dirige le système
- Téléopérateur : le système dirigé
- Envoi de commande via manettes, interfaces ou capteurs
- Mouvements retransmis au système par l’intermédiaire de signaux directs, rétroaction en temps réel : commande en boucle fermé
- Divers systèmes opérés : avatar, robot, humanoïde, animal, ...



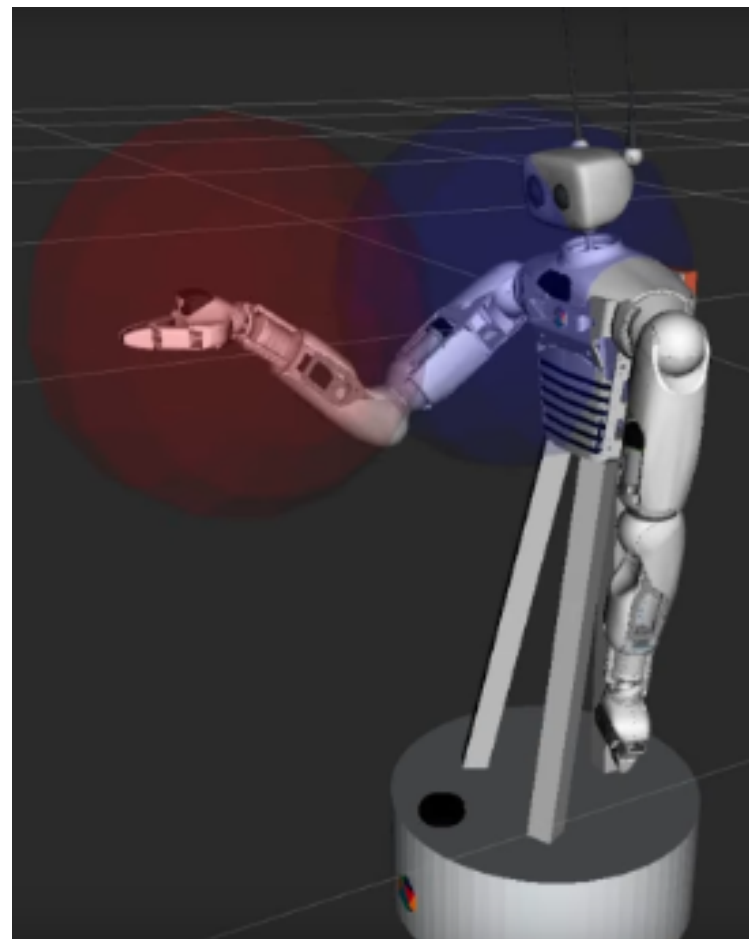
Solutions multiples

La cinématique inverse peut avoir une, plusieurs, ou aucune solution selon la position cible.



Représentation des
solutions atteignables pour
un robot plan 2R

Représentation des
solutions atteignables pour
un robot 7R



Butées articulaires et singularités

Perte de mobilités en atteignant les limites articulaires ou lors des singularités.

$$J = \underbrace{U}_{m \times m} \underbrace{\Sigma}_{m \times n} \underbrace{V^T}_{n \times n}$$

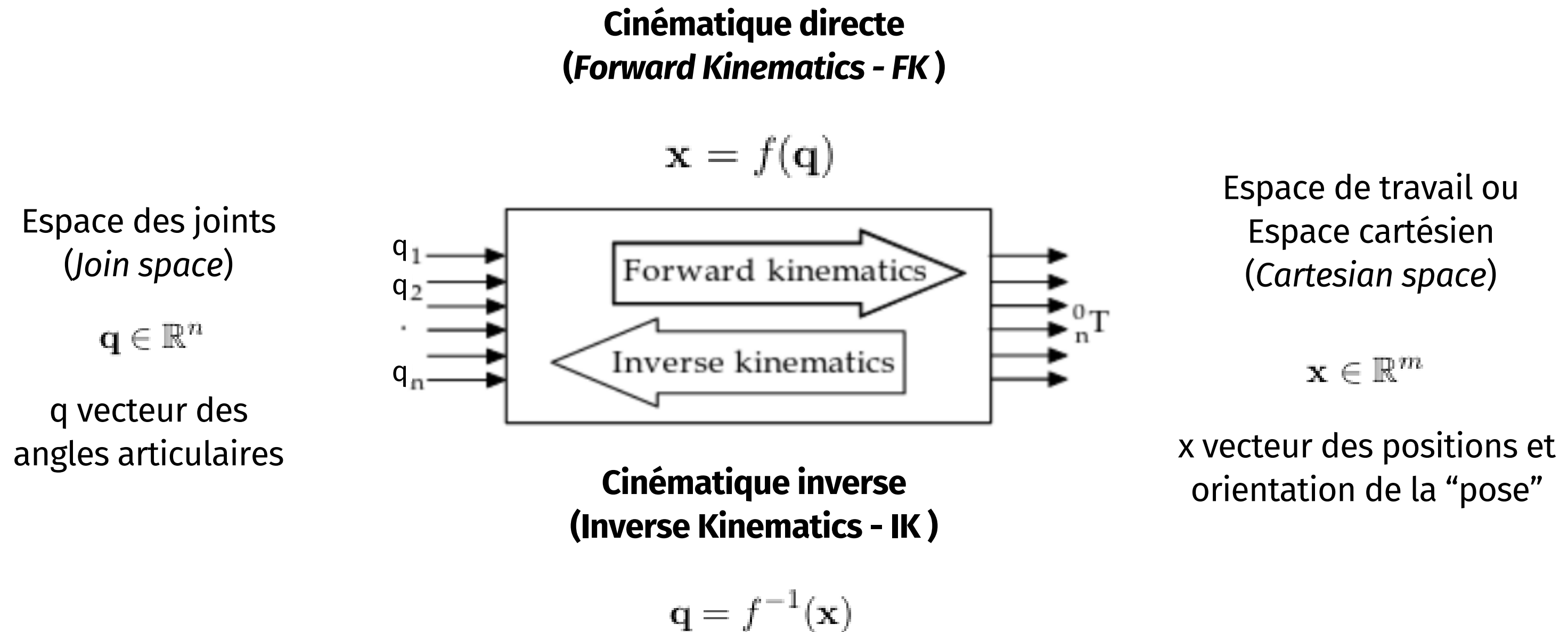
Σ is a diagonal matrix of singular values:

$$J = U \begin{bmatrix} \sigma_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_n & 0 & 0 \end{bmatrix} V^T$$

$$J^\# = V \Sigma^{-1} U^T$$

$$J^\# = V \begin{bmatrix} \frac{1}{\sigma_1} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\sigma_2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sigma_3} & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sigma_n} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} U^T$$

Écriture d'un problème de cinématique



Méthodes Analytiques

Méthodes analytiques :

- Méthodes algébriques
- Méthodes géométriques

--> Calculs formels

--> Entrées : x_d , les longueurs des segments de la structure.

--> Sortie : q_d

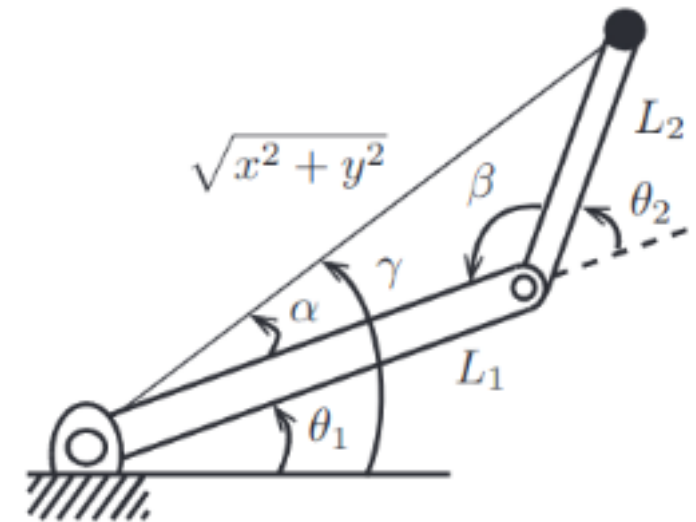


Schéma du problème de cinématique inverse pour un robot plan 2R [35]

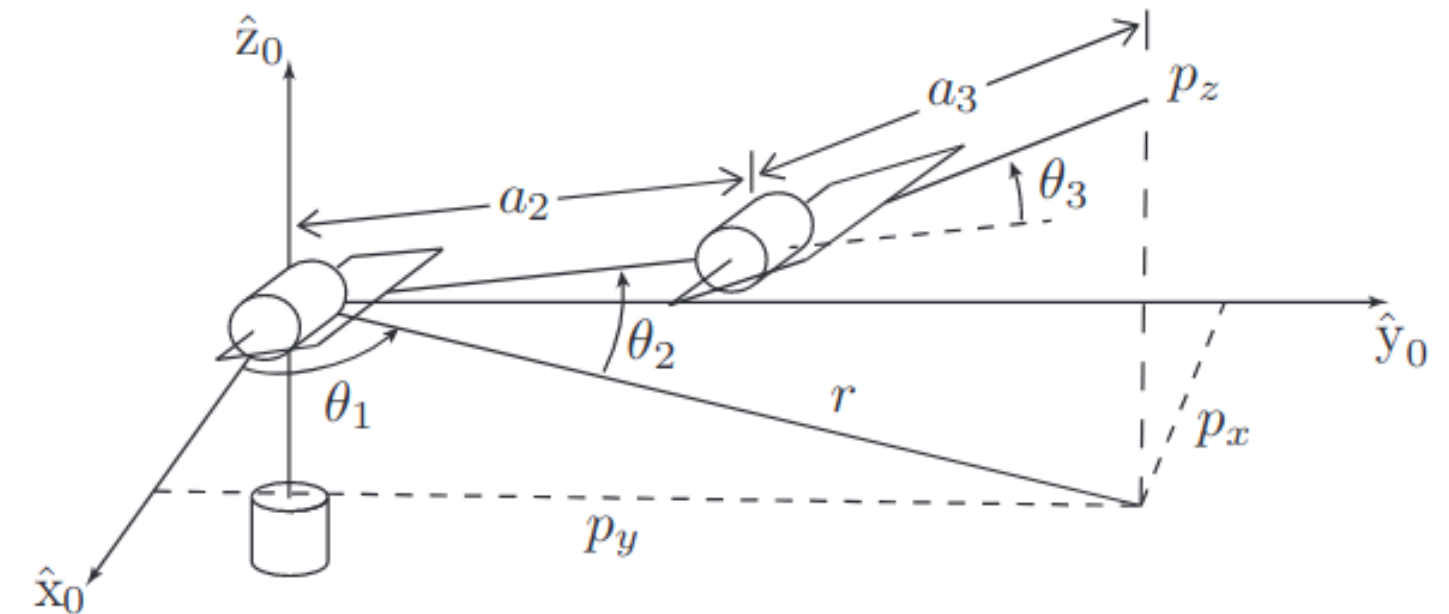
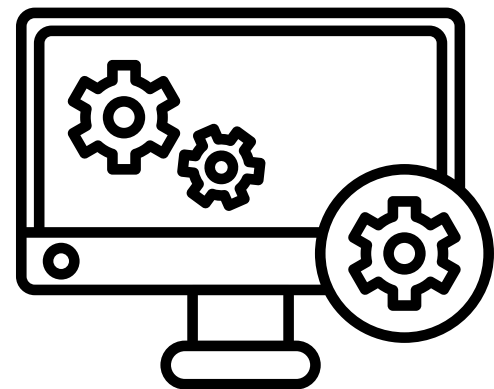


Schéma du problème de cinématique inverse pour un bras 6R type PUMA



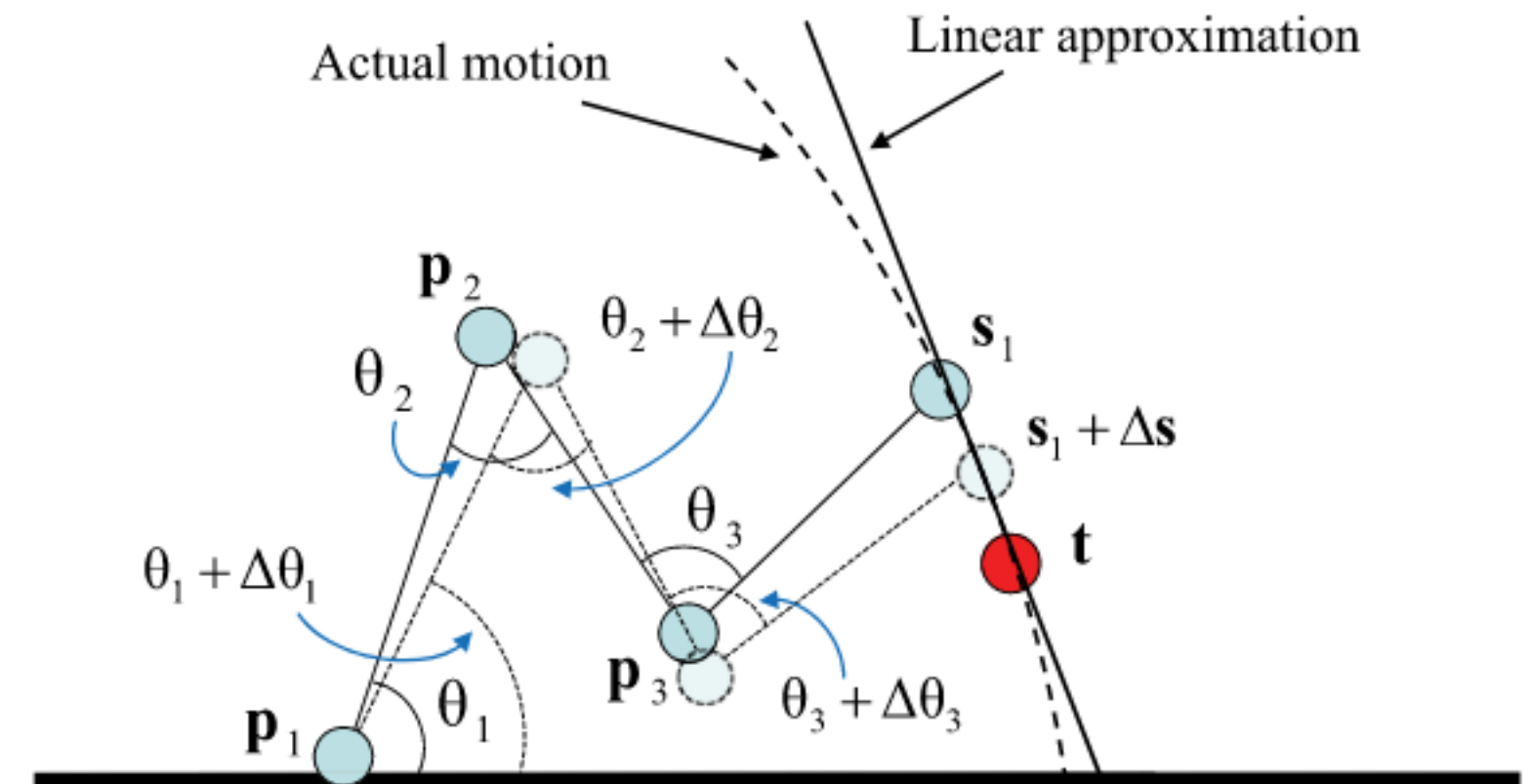
Résoudre en programmation

- Bibliothèques pour le calcul symbolique : **SymPy** en Python
- **SageMath**, une alternative *open-source* des logiciels **Maple**, **Magma** ou **Mathematica**.

Méthodes Numériques

Méthodes numériques :

- Méthodes itératives
 - Newton-Raphson
 - Méthode de la bisection
 - Méthode de la sécante
- Méthodes utilisant les dérivées
 - Jacobienne transposée
 - Pseudo-inverse jacobienne
- Méthodes hybrides
 - Moindre carrés amortis (DLS)
 - Algorithme de cinématique inverse en boucle fermée (CLIK)
- Méthodes d'optimisation
 - LP, QP, SOCP, SDP, ...

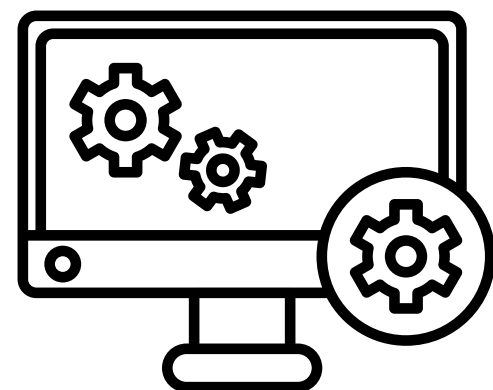


Résolution de la cinématique inverse par une approche itérative (Jacobienne)

--> Approximation de la solution

--> Entrée : $q_0, x_d, f(q), J(q)$

--> En sortie : q_d , erreur résiduelle



Résoudre en programmation

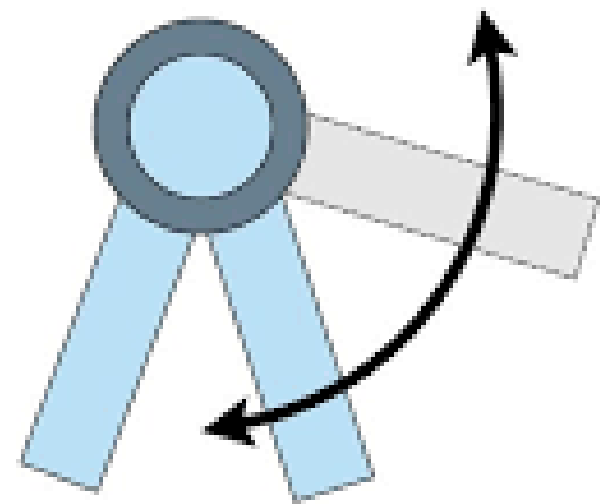
- Bibliothèques **NumPy** et **SciPy** de Python ou encore **CasADi**, un outil *open-source* pour l'optimisation non linéaire
- **Pinocchio** *open source* efficace pour calculer la dynamique de corps articulés comme un bras

Récapitulatifs

	Méthodes analytiques	Méthodes numériques
Avantages	<ul style="list-style-type: none">- Solution exacte, sans approximation- Pas de dépendance aux conditions initiales	<ul style="list-style-type: none">- Flexibilité : Applicable à une large gamme de robot- Adaptabilité : peut intégrer des contraintes d'environnement- Gestion de la redondance avec des critères d'optimisations
Inconvénients	<ul style="list-style-type: none">- Complexe pour certains systèmes- Rigide, n'est pas adapté dans des conditions changeantes- Ne comprend pas le calcul de la dérivé	<ul style="list-style-type: none">- La solution est une estimation- Le temps de calcul peut être long- Dépend des conditions initiales

Comparaison des méthodes analytiques et itératives pour la résolution de problèmes de cinématique inverse.

Une contrainte



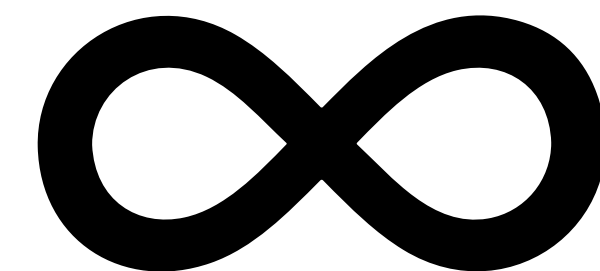
Butée articulaire



Collision



Tâches



et plus encore

Ajout de
contraintes

***Explicit inversion
methods***

$$\dot{\mathbf{q}}^c = J^\dagger(\mathbf{q})\dot{\mathbf{x}}_d + \left(I_n - J^\dagger(\mathbf{q})J(\mathbf{q}) \right) \dot{\mathbf{q}}_0$$

***Constrained convex
optimization methods***

$$\begin{aligned} &\text{minimiser } f(\mathbf{y}) \\ &\text{avec } h(\mathbf{y}) \leq 0 \\ &\quad g(\mathbf{y}) = 0 \end{aligned}$$

Explicit inversion methods

$$\dot{\mathbf{q}}^c = J^\dagger(\mathbf{q})\dot{\mathbf{x}}_d + \left(I_n - J^\dagger(\mathbf{q})J(\mathbf{q})\right)\dot{\mathbf{q}}_0$$

Contrainte = Sous-tâche

Degrés de redondance

└ Degrés de liberté
de sous-tâche

Généralisation à n tâches :

$$\begin{pmatrix} \dot{x}_{d,1} \\ \dot{x}_{d,2} \\ \dots \\ \dot{x}_{d,n} \end{pmatrix} = \begin{pmatrix} J_1 \\ J_2 \\ \dots \\ J_n \end{pmatrix} \dot{\mathbf{q}}$$

Résolution :

$$\dot{\mathbf{q}}_i^c = \dot{\mathbf{q}}_{i-1}^c + \left(J_i(I_n - J_{i-1}^\dagger J_{i-1})\right)^\dagger (\dot{\mathbf{x}}_{d,i} - J_i \dot{\mathbf{q}}_{i-1})$$

Explicit inversion methods



Hiérarchisation de tâche
Pas d'interférence entre les tâches



Si aucun degrés de disponible
↳ Non exécution des tâches de rang inférieur
Uniquement les contraintes articulaire

Constrained convex optimization methods

$$f(\mathbf{y}) = ||E\mathbf{y} - \mathbf{f}||_2^2 \longrightarrow \mathbf{y} = \dot{\mathbf{q}} \quad \mathbf{f} = \dot{\mathbf{x}} \quad E = J$$

$$||J\dot{\mathbf{q}} - \dot{\mathbf{x}}||_2^2 = (J\dot{\mathbf{q}} - \dot{\mathbf{x}})^T (J\dot{\mathbf{q}} - \dot{\mathbf{x}}) = \dot{\mathbf{q}}^T J^T J \dot{\mathbf{q}} - 2\dot{\mathbf{x}}^T J \dot{\mathbf{q}} + \dot{\mathbf{x}}^T \dot{\mathbf{x}}$$

$$\dot{\mathbf{q}}^{opt} = \arg \min_{\dot{\mathbf{q}}} \dot{\mathbf{q}}^T H \dot{\mathbf{q}} - 2\nabla \dot{\mathbf{x}}(\dot{\mathbf{q}})^T \dot{\mathbf{q}} + ||\dot{\mathbf{x}}||_2^2$$

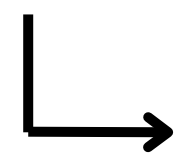
s.t.

$$\mathbf{l}_b \leq A\dot{\mathbf{q}} \leq \mathbf{u}_b$$

$$C\dot{\mathbf{q}} = \mathbf{d}$$

Constrained convex optimization methods

Résolution



Méthode d'optimisation quadratique



Quantification de l'importance des contraintes
Pas de spécification de contrainte
Prise en compte des tâches et des contraintes



Opération complexe
Temps de calcul rallongé

Méthodes d'optimisation et solveurs

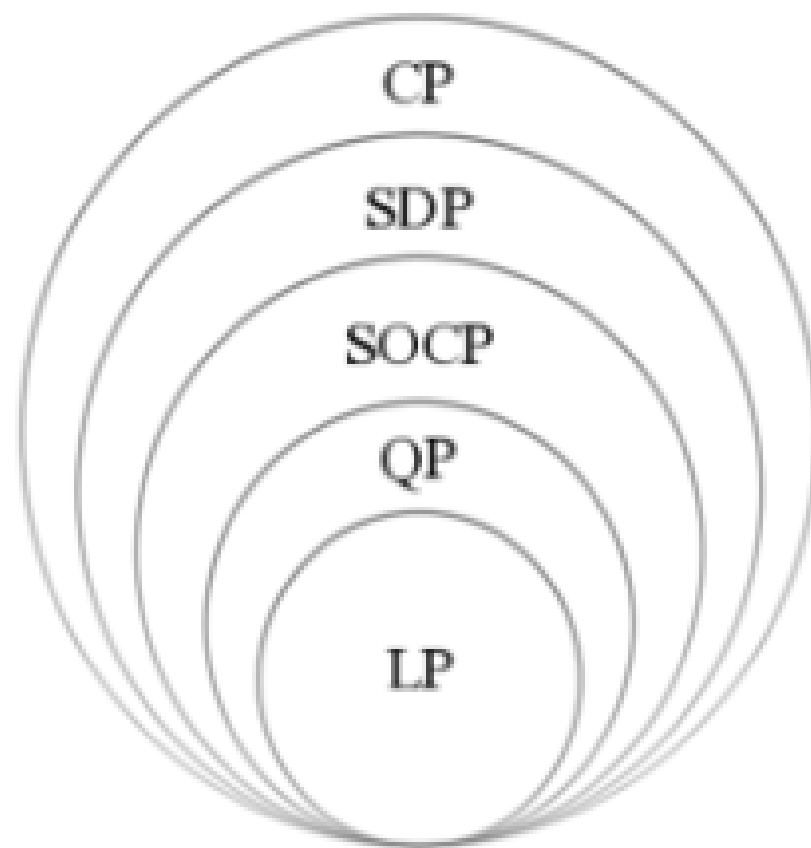
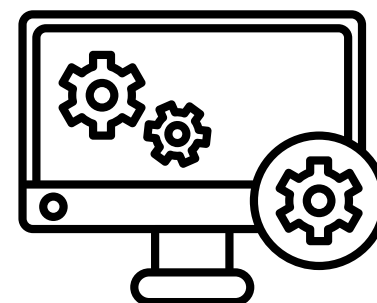


Schéma récapitulatif des méthodes d'optimisation convexe

Méthodes d'optimisation :

- LP (*Linear programming*)
- QP (*Quadratic programming*)
- SOCP (*Second Order Cone Programming*)
- SDP (*Semidefinite programming*)
- CP (*Conic optimization*)

➔ La fonction objectif ne permettent pas l'usage de certaines méthodes (LP, SDP).



Résoudre en programmation

Outils Python, Pink (basé sur Pinocchio), CVXOPT, Quadprog, Pyomo, IPOPT, OSQP, CVXPY.

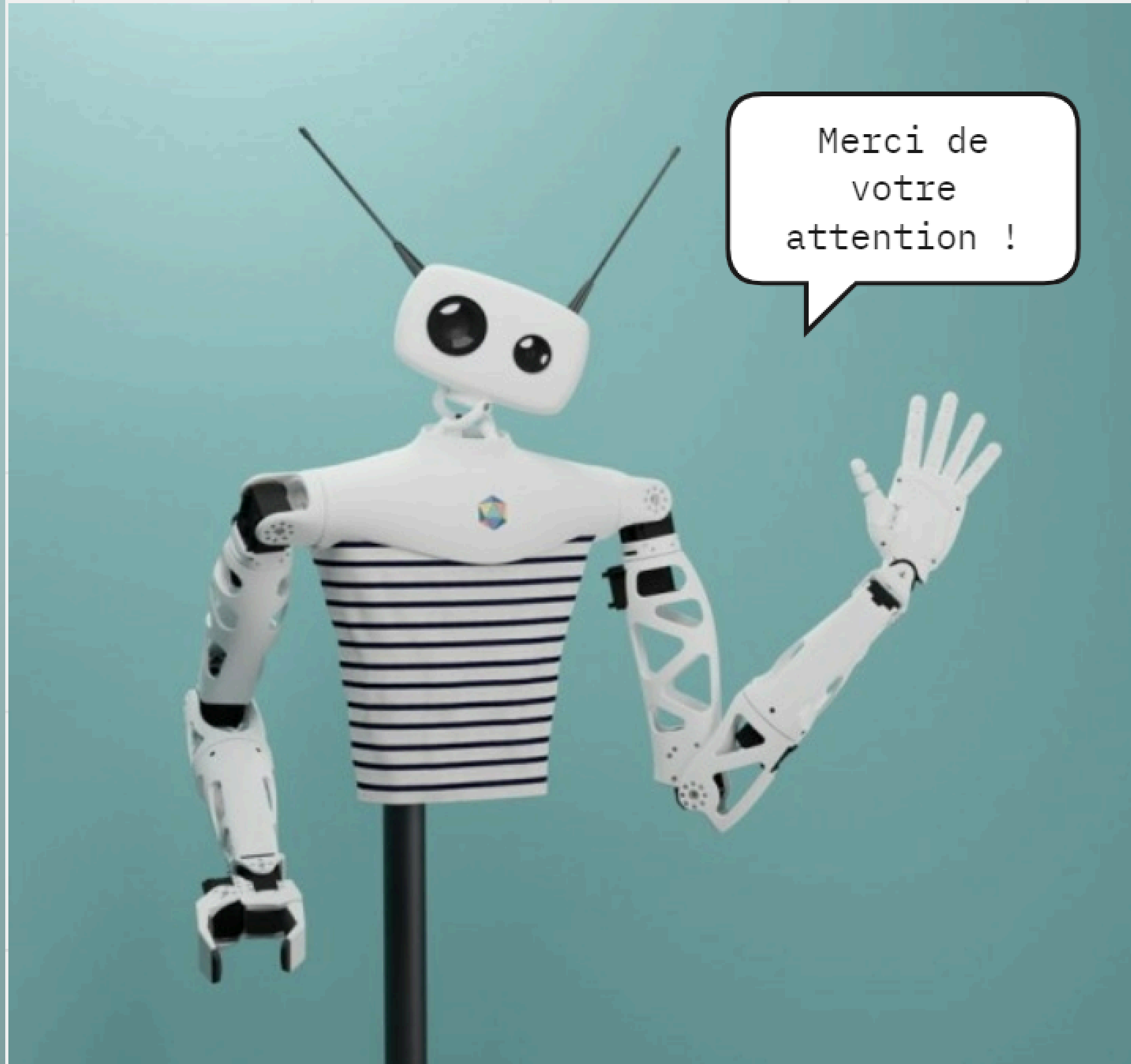
Conclusion

- Identification des problèmes rencontrés en téléopération
- Comparaison des méthodes de résolution de la cinématique inverse
- Outils pour une résolution en programmation



Suite du projet

- Utilisation méthode Jacobienne
- Implémentation QP
- Mise en place de contraintes
- Evaluation



Bibliographie

- [1] Akshay Agrawal, Robin Verschueren, Steven Diamond, and Stephen Boyd. A rewriting system for convex optimization problems. 5(1) :42–60.
- [2] Martin S. Andersen, Joachim Dahl, and Lieven Vandenbergh. Solving a quadratic program — CVXOPT.
- [3] Joel A. E. Andersson, Joris Gillis, Greg Horn, James B. Rawlings, and Moritz Diehl. CasADi : a software framework for nonlinear optimization and optimal control. 11(1) :1–36.
- [4] A. Aristidou, J. Lasenby, Y. Chrysanthou, and A. Shamir. Inverse kinematics techniques in computer graphics : A survey. 37(6) :35–58.
- [5] Andreas Aristidou and Joan Lasenby. Inverse kinematics : a review of existing techniques and introduction of a new fast iterative solver.
- [6] Laura Aymerich-Franch, Damien Petit, Gowrishankar Ganesh, and Abderrahmane Kheddar. Non-human looking robot arms induce illusion of embodiment. 9(4) :479–490.
- [7] Antal K. Bejczy. Sensors, controls, and man-machine interface for advanced teleoperation. 208(4450) :1327–1335.
- [8] Michael J. Best and Klaus Ritter. Linear programming : active set analysis and computer programs. Prentice-Hall.
- [9] Immanuel M. Bomze. On standard quadratic optimization problems. 13(4) :369–387.
- [10] Eli Bressert. SciPy and NumPy : An Overview for Developers. "O'Reilly Media, Inc."
- [11] Stéphane Caron, Yann De Mont-Marin, Rohan Budhiraja, Seung Hyeon Bang, Ivan Domrachev, and Simeon Nedelchev. Pink : Python inverse kinematics based on pinocchio. original-date : 2022-02-06T10 :05 :13Z.
- [12] Justin Carpentier, Guilhem Saurel, Gabriele Buondonno, Joseph Mirabel, Florent Lamiriaux, Olivier Stasse, and Nicolas Mansard. The pinocchio c++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives.
- [13] Justin Carpentier, Guilhem Saurel, Gabriele Buondonno, Joseph Mirabel, Florent Lamiriaux, Olivier Stasse, and Nicolas Mansard. pinocchio : Inverse kinematics (klik).
- [14] Pasquale Chiacchio, Stefano Chiaverini, Lorenzo Sciavicco, and Bruno Siciliano. Closed-loop inverse kinematics schemes for constrained redundant manipulators with task space augmentation and task priority strategy. 10(4) :410–425.
- [15] Steven Diamond and Stephen Boyd. CVXPY : A python-embedded modeling language for convex optimization.
- [16] A. D'Souza, S. Vijayakumar, and S. Schaal. Learning inverse kinematics. In Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180), volume 1, pages 298–303 vol.1.
- [17] Thomas S Fergusin and M Casquilho. Linear programming, a concise introduction.
- [18] Rebecca Fribourg, Evan Blanpied, Ludovic Hoyet, Anatole Lécuyer, and Ferran Argelaguet. Does virtual threat harm VR experience ? : Impact of threat occurrence and repeatability on virtual embodiment and threat response. 100 :125.
- [19] D. Goldfarb and A. Idnani. A numerically stable dual method for solving strictly convex quadratic programs. 27(1) :1–33.
- [20] Computational Algebra Group, Sydney School of Mathematics {and} Statistics, and University of Sydney. Magma computational algebra system.

Bibliographie

- [21] William E. Hart, Carl D. Laird, Jean-Paul Watson, David L. Woodruff, Gabriel A. Hackebeil, Bethany L. Nicholson, and John D. Sirola. Pyomo — Optimization Modeling in Python, volume 67 of Springer Optimization and Its Applications. Springer International Publishing.
- [22] Ahmed A. Hassan, Mohamed El-Habrouk, and Samir Deghedie. Inverse kinematics of redundant manipulators formulated as quadratic programming optimization problem solved using recurrent neural networks : A review. 38(8) :1495–1512.
- [23] Didier Henrion and Jérôme Malick. Projection methods in conic optimization.
- [24] M. Hiller. Multiloop kinematic chains. In J. Angeles and A. Kecskeméthy, editors, Kinematics and Dynamics of Multi-Body Systems, pages 75–165. Springer.
- [25] Lucas Joseph. An energetic approach to safety in robotic manipulation.
- [26] Oussama Kanoun. Real-time prioritized kinematic control under inequality constraints for redundant manipulators. In Robotics : Science and Systems VII. Robotics : Science and Systems Foundation.
- [27] Oussama Kanoun, Florent Lamiroux, and Pierre-Brice Wieber. Kinematic control of redundant manipulators : Generalizing the task-priority framework to inequality task. 27(4) :785–792.
- [28] Sunyoung Kim and Masakazu Kojima. Exact solutions of some nonconvex quadratic optimization problems via SDP and SOCP relaxations.
- [29] Nikos Kofinas, Emmanouil Orfanoudakis, and Michail G. Lagoudakis. Complete analytical inverse kinematics for NAO. In 2013 13th International Conference on Autonomous Robot Systems, pages 1–6. IEEE.
- [30] Andrey Krekhov, Sebastian Cmentowski, and Jens Krüger. The illusion of animal body ownership and its potential for virtual reality games. In 2019 IEEE Conference on Games (CoG), pages 1–8. ISSN : 2325-4289.
- [31] Serdar Kucuk and Zafer Bingul. Robot kinematics : Forward and inverse kinematics. In Sam Cubero, editor, Industrial Robotics : Theory, Modelling and Control. Pro Literatur Verlag, Germany / ARS, Austria.
- [32] S Lichiardopol. A survey on teleoperation. page 34.
- [33] Weihui Liu, Diansheng Chen, and Jochen Steil. Analytical inverse kinematics solver for anthropomorphic 7-DOF redundant manipulators with human-like configuration constraints. 86(1) :63–79.
- [34] Alain Liégeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. 7(12) :868–871. Conference Name : IEEE Transactions on Systems, Man, and Cybernetics.
- [35] Kevin M. Lynch and Frank C. Park. Modern Robotics : Mechanics, Planning, and Control. Cambridge University Press, 1 edition.
- [36] Anthony Maciejewski and Charles Klein. Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. 4.
- [37] Athanasios Makrodimopoulos and Chris Martin. Limit analysis using large-scale SOCP optimization.
- [38] Waterloo Maple. Maple - l'outil essentiel pour les mathématiques et la modélisation - maplesoft.
- [39] Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, Štěpán Roučka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. SymPy : symbolic computing in python. 3 :e103.

Bibliographie

- [40] N. Miller, O.C. Jenkins, M. Kallmann, and M.J. Mataric. Motion capture from inertial sensing for untethered humanoid teleoperation. In 4th IEEE/RAS International Conference on Humanoid Robots, 2004., volume 2, pages 547–565. IEEE.
- [41] Pollen Robotics. Robot kinematics explained.
- [42] Pollen Robotics. Pollen robotics.
- [43] Pollen Robotics. Reachy 2023 documentation.
- [44] Akram Saba and ul Ann Qurrat. Newton raphson method. 6 :1748–1752.
- [45] Philip N Sabes. Linear algebraic equations, SVD, and the pseudo-inverse.
- [46] Pollen Robotics SAS. Reachy, developed by pollen robotics, is an open-source humanoid robot.
- [47] Bruno Siciliano and J.-J.E. Slotine. A general framework for managing multiple tasks in highly redundant robotic systems. In Fifth international conference on advanced robotics (Vol. 2. Pages : 1216 vol.2.
- [48] Bartolomeo Stellato, Goran Banjac, Paul Goulart, Alberto Bemporad, and Stephen Boyd. OSQP : an operator splitting solver for quadratic programs. 12(4) :637–672.
- [49] Pragati Thapliyal and Komal Tomar. Chitra solanki DIT university dehradun, india. 3(8)
- [50] The SageMath Developers. SageMath.
- [51] Alexander Toet, Irene A. Kuling, Bouke N. Krom, and Jan B. F. van Erp. Toward enhanced teleoperation through embodiment. 7.
- [52] Hamid Toshani and Mohammad Farrokhi. Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming : A lyapunov-based approach. 62(6) :766–781.
- [53] Lieven Vandenberghe and Stephen Boyd. Semidefinite programming. 38(1) :49–95. Publisher : Society for Industrial and Applied Mathematics.
- [54] Jingguo Wang, Yangmin Li, and Xinhua Zhao. Inverse kinematics and control of a 7-DOF redundant manipulator based on the closed-loop algorithm. 7(4) :37. Publisher : SAGE Publications.Bibliographie
- [55] Philip Wolfe. The secant method for simultaneous nonlinear equations. 2(12) :12–13.
- [56] Stephen Wolfram. Wolfram mathematica : le calcul technique et moderne.
- [57] Andreas Wächter and Carl Laird. coin-or/ipopt. original-date : 2019-01-28T12 :24 :22Z.