

# Kinematics and dynamics motion planning by polar piecewise interpolation and geometric considerations

Mihai Dupac<sup>1</sup>

*Faculty of Science and Technology  
Bournemouth University  
Poole, United Kingdom*

---

## **Abstract**

The importance of numerical methods in science and engineering[1] was long recognised and considered a fundamental factor in improving productivity and reducing production costs. The ability to model flexible systems and describe their trajectories [5] involves usually the study of nonlinear coupled partial differential equations. Since their exact solutions are not normally feasible in practice, computational methods [8] can be considered.

*Keywords:* Numerical methods, motion planning, kinematics, dynamics.

---

The development of such computationally efficient numerical algorithms and methods for solving PDE and handling of complicated geometries [13] require the use of implicit (trajectory and forces computed along the motion) and explicit (trajectory computed before execution) motion planning methods [14]. Implicit approaches based on potential field methods consider a potential

---

<sup>1</sup> Email: mdupac@bournemouth.ac.uk

function in the C-space to asymptotically generate an attractive goal configuration. The potential function can be defined [8] by

$$(1) \quad U(q) = \frac{1}{2}\zeta\|q - q_{Goal}\|$$

where  $\zeta$  is a scale factor. Explicit methods [8] can be separate in continuous methods namely kinematic or dynamic motion planning [16] or discrete methods. The kinematic approach often referred as *geometry of motion* is essential the initial step in the computation of the forces related to the dynamical model (dynamical equations of the system) and their effect on the motion. Kinematic motion planning [8] formulated as a variational approach consist in generating a smooth trajectory  $\delta : [0, 1] \rightarrow \mathbb{R}$  by minimizing a functional

$$(2) \quad J(\delta) = \int_0^1 \left\langle L\left(\delta, \frac{d\delta}{dt}\right), L\left(\delta, \frac{d\delta}{dt}\right) \right\rangle dt, \quad \text{and}$$

where  $t$  is the integration time,  $\langle \cdot, \cdot \rangle$  is the Riemannian metric and  $L$  the initial conditions [8]. Dynamic motion planning [8] can be defined as a mathematical minimization of a cost function [16] by

$$(3) \quad \min(J_c) \quad \text{where} \quad J_c = \Phi(y(t_f)) + \int_{t_0}^{t_f} L(y, u) dt$$

subject to initial conditions  $y(t_0) = x_0$  and  $\dot{y} = f(y, u)$ . Parameterizations of joint trajectories transform the dynamic motion planning problem into a parameter optimization [8] problem of a family of smooth curves generated by parametric and/or geometric continuous splines, uniform cubic  $B$  spline with parametric and geometric continuity, or by piecewise interpolating curves [4,6,7,9].

Since the motion along a set of control point can be eventually expressed as a combination of rotations about a fixed point, that is, rotations about  $Ox$ ,  $Oy$  and  $Oz$  axes that passes through the origin, the use of polar and/or Cartesian piecewise interpolating curves [2,11] can be then related with the robot dynamics [2,10,12] through its geometry [2]. The set of the considered rotations can be represented as a 3D manifold, that is, the real projective 3-space denoted by  $RP^3$ . Depending on the geometry of a robotic manipulator arm, the fixed point representing the center of rotation, e.g., base of the manipulator, can be related to either polar zenithal gnomonic, polar zenithal stereographic or polar zenithal orthographic projections, by the use of the time dependent generalized coordinates expressed as spherical or cylindrical coordinates. To interpolate between some a priori defined control points, a piecewise

polar interpolant that approximate the polar trajectory can be expressed as a Hermite-type function [2,7] defined by

$$(4) \quad r(\theta) = \sum_{k=0}^q c_k^i (\theta - \theta_i)^k$$

where  $q = 3$  (cubic approximation) is the order of the polynomial,  $c_0^i = r_i$ ,  $c_1^i = \dot{r}_i$ ,  $c_2^i = \frac{1}{h_i} [(2\dot{r}_i + \dot{r}_{i+1}) + 3\Delta r_i]$ ,  $c_3^i = \frac{1}{h_i^2} [\dot{r}_i + \dot{r}_{i+1} - 2\Delta r_i]$ ,  $r_i = r(\theta_i)$ ,  $r_{i+1} = r(\theta_{i+1})$ ,  $h_i = \theta_{i+1} - \theta_i$ ,  $\Delta r_i = \frac{r_{i+1} - r_i}{h_i}$ , and the derivatives at the endpoints are calculated using  $\dot{r}(\theta_i) = \frac{dr(\theta_i)}{d\theta} = \dot{r}_i$  and  $\dot{r}(\theta_{i+1}) = \frac{dr(\theta_{i+1})}{d\theta} = \dot{r}_{i+1}$  respectively.

Considering the case of polar zenithal orthographic projection and related cylindrical coordinates (Fig. 1.c), trajectories representing the system dynamics can be generated using a combination of polar piecewise interpolants that approximate the polar trajectory, and Cartesian piecewise interpolants to approximate the trajectory height. Trajectory height is computed in the projective plane obtained by unfolding the cylinder with generators passing through the polar trajectory of the motion.

Considering the case of polar gnomonic or stereographic projection and related spherical coordinates (Fig. 1.a and Fig. 1.b), trajectories representing the system dynamics can be generated by a combination of polar piecewise interpolants over a ‘‘spherical’’ domain relatively similar with [6] by

$$(5) \quad R(\lambda, \phi) = S_0(\psi) + S_1(\psi)\chi + S_2(\psi)\chi^2 + S_3(\psi)\chi^3$$

where  $\lambda$  and  $\phi$  represents the generalised coordinates related to the system model,  $S(\psi) = c_{0,0} + c_{1,0}\psi + c_{2,0}\psi^2 + c_{3,0}\psi^3$ , and  $\psi$  and  $\chi$  are defined as in [6].

Therefore, adequate geometric considerations - such as manipulators geometry and related coordinate systems - used in the motion planning studies backed up by polar piecewise cubic interpolants allows easy calculation of kinematics variable and handling of the dynamical equations of motion [12] through the use of Lagrange’s equations defined by

$$(6) \quad \frac{d}{dt} \left( \frac{\partial T}{\partial \dot{q}_j} \right) - \frac{\partial T}{\partial q_j} = Q_j$$

The kinetic energy  $T$  = and the acting contact and body forces  $Q_i$  can be expressed as in [12] by

$$(7) \quad T_j = \frac{1}{2} m_j \mathbf{v}_{C_j} \cdot \mathbf{v}_{C_j} + \frac{1}{2} \bar{I}_j \omega_j \cdot \omega_j$$

and

$$(8) \quad Q_j = \frac{\partial \omega}{\partial \dot{q}_j} \cdot \mathbf{T} + \frac{\partial \mathbf{v}}{\partial \dot{q}_j} \cdot \mathbf{R}$$

where  $v$  is the velocity,  $\omega$  is the angular velocity,  $\mathbf{T}$  is the applied torque, and  $\mathbf{R}$  is the applied force.

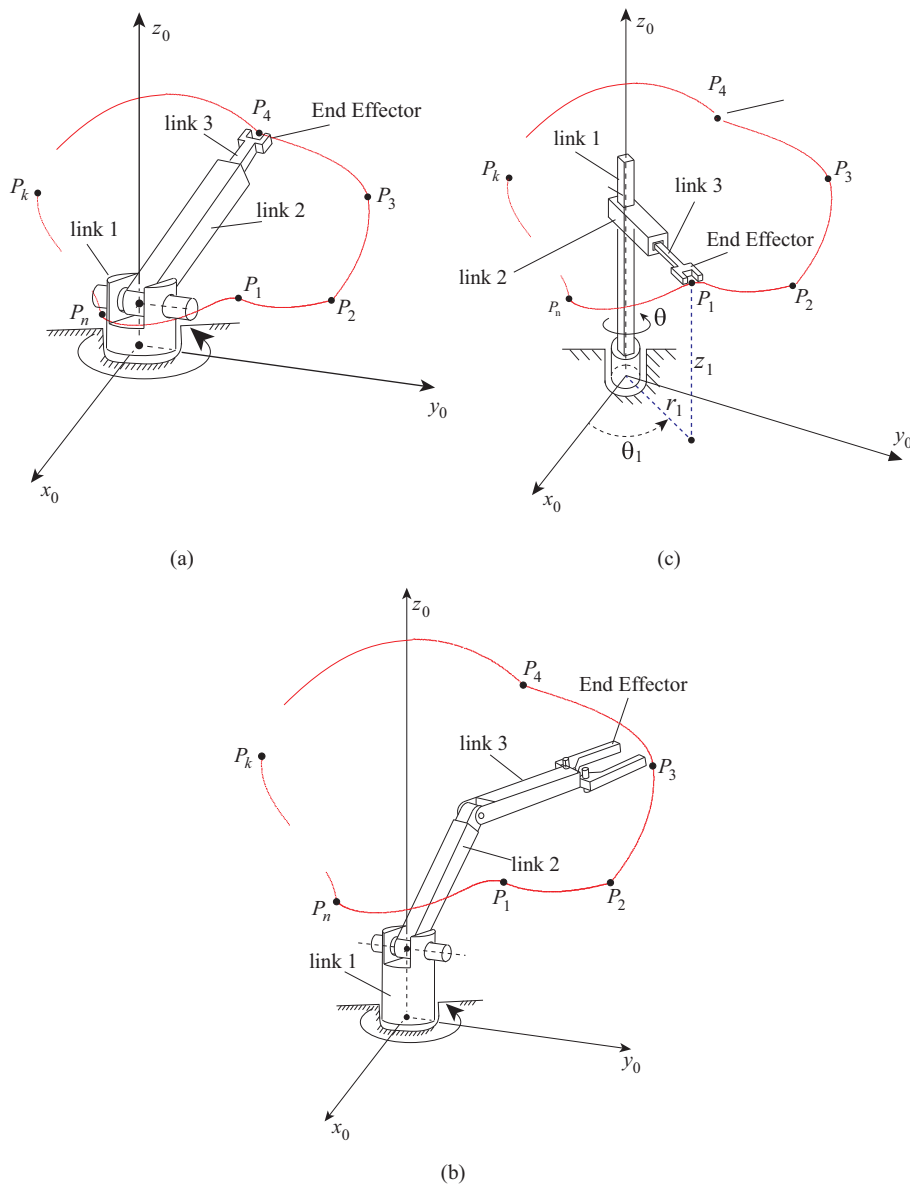


Figure 1

## References

- [1] Chapra, S.C., and R.P. Canale, “Numerical Methods for Engineers,” McGraw-Hill, 6th Ed., 2010.
- [2] Dupac M., *Smooth trajectory generation for rotating extensible manipulators*, Mathematical Methods in Applied Sciences, 2016.
- [3] Martinez Garcia, E.A., “Numerical modeling in Robotics,” Omnia Science, 2015.
- [4] Goodman, T.N.T., and S.L. Lee , *B-splines on the Circle and Trigonometric B-splines*, Approximation Theory and Spline Functions **136** (1984), pp. 297–325.
- [5] Gasparetto A, Boscaroli P, Lanzutti A, Vidoni R., *Trajectory planning in Robotics*, Mathematics in Computer Science **6** (2012), pp. 269–279.
- [6] M. Gross, A. Staniforth, *Cubic-spline interpolation on a non-uniform latitude-longitude grid: achieving cross- and circum-polar continuity*, Atmos. Sci. Let. **11** (2010), pp. 229–23.
- [7] Iwashita Y. *Piecewise Polynomial interpolation*. OpenGamma Quantitative Research **15** (2014), pp. 1-22.
- [8] V. Kumar, M. Zefran, J.P. Ostrowski, *Motion Planning and Control of Robots*, Handbook of Industrial Robotics, 2nd Edition, J. Wiley and Sons (2007), pp. 295–315.
- [9] Kang IG, Park FC. *Cubic Spline Algorithms for Orientation Interpolation*, International Journal for Numerical Methods in Engineering, **46** (1999) pp. 45–64.
- [10] Kalyoncu M. *Mathematical modelling and dynamic response of a multistraight-line path tracing flexible robot manipulator with rotating-prismatic joint*, Applied Mathematical Modelling **32** (2008), pp. 1087-1098.
- [11] J.E. Lavery, *Shape-preserving, multiscale interpolation by univariate curvature-based cubic L1 splines in Cartesian and polar coordinates*, Computer Aided Geometric Design **19** (2002) pp. 257–273.
- [12] Marghitu DB, Dupac M. “Advanced Dynamics: Analytical and Numerical Calculations with Matlab,” Springer: New York, 2012.
- [13] R.M. Murray, Z. Li, S.S. Sastry, “A Mathematical Introduction to Robotic Manipulation,” CRC Press, 1994.

- [14] LJ du Plessis, Snyman JA., *Trajectory-planning through interpolation by overlapping cubic arcs and cubic splines*, International Journal for Numerical Methods in Engineering **57** (2003), pp. 1615-1641.
- [15] S. K. Dwivedy, P. Eberhard, *Dynamic analysis of flexible manipulators, a literature review*, Mechanism and Machine Theory. **41** (2006), pp. 749–777.
- [16] C.-W.E. Wang, “Dynamic Motion Planning For Robot Manipulators Using B-Splines,” PhD Dissertation, University of California, Irvine, (2001), pp. 1–155.