

Trajectory optimization for redundant/hyper redundant manipulators

Kagan Koray Ayten*, Ahmet Dumlu

Department of Electrical & Electronics Engineering, College of Engineering, Erzurum Technical University, Erzurum, Turkey



ARTICLE INFO

Article history:

Received 28 February 2017

Received in revised form

17 September 2017

Accepted 21 September 2017

Keywords:

Trajectory planning

Energy minimization

Constraint handling

Cost function

Hyper-redundant manipulators

Redundant manipulator

ABSTRACT

This paper presents an optimization technique to generate minimum energy consumption trajectories for redundant/hyper-redundant manipulators with constraint conditions. Two novel methods are combined for trajectory optimization technique. In the first method, the system's constraints are handled in a sequential manner within the cost function in order to avoid running the inverse dynamics when the system's constraints are not satisfied. In the second method, a novel virtual link concept is introduced to replace all the redundant links of the redundant/hyper-redundant manipulators. The method is verified on a three-degree-of-freedom redundant manipulator and the result is also demonstrated with computer simulations based on an 8-link planar hyper-redundant manipulator.

© 2017 The Authors. Published by IASE. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

Although the non-redundant commercial robot's performance and capability provide significant advantages for industrial implementations, however, in today's modern industrial world consist of multi-task industrial issues. The requirements of industrial applications are vastly complex, difficult and these applications demand better performance and flexibility such as drilling, cutting, medical robotics, maintenance of nuclear reactors etc. In order to meet these demands, robotic manipulators may have more degrees-of-freedom than essential due to execute intended complicated jobs like human arms. Research on redundant robotic manipulator is still active and redundancy can also be utilised to handle other cost criteria effectively. For example, redundancy has been used to achieve collision avoidance in working space (Patel et al., 2005; Ping et al., 2009), preventing singularities where the manipulator lose some degrees-of-freedom (Duarte et al., 1999; Huo and Boron, 2008), avoiding limits of joints (Assal et al., 2006; Marey and Chaumette, 2010), minimizing some cost function such as reducing jerk (Sullivan and Pipe, 1998; Yang et al., 2008), reducing deviation of the end-effectors (Ata and Myo, 2005; Singla et al., 2010), reducing time (Ma and Watanabe, 2000) over an intended task.

This paper presents a novel constraint handling method and an efficient control algorithm for trajectory optimization to prevent computational complexity of the redundant and hyper-redundant manipulators. One of the main contributions of this paper is to provide constraint handling procedure is computationally efficient as kinematic and dynamic constraints are included in the cost function to prevent running inverse dynamic model when all constraints are not satisfied. And second contribution of this paper, all the redundant links are acting as a single link. The effectiveness of the proposed methods is initially demonstrated using computer simulations and then same intended trajectories are implemented experimentally by utilising the Katana 450 6M industrial robotic manipulator based on links 2, 3 and 4. The proposed scheme is also verified with computer simulations based on an 8-link planar hyper-redundant manipulator.

2. System description and dynamics

Fig. 1 shows the six degrees-of-freedom of the Katana 450 robotic manipulators with end-effector tool. Dynamic modelling of the robot is based on Lagrangian dynamics, which describes the system in terms of its energy. To construct the inverse dynamic model of the system, the DYSIM software is utilized (Sahinkaya, 2004) and it will operate from within the MatLab/Simulink environment. DYSIM requires the user to specify the mass, inertia and position of centre of mass for each link, as well as identifying the location of ground point. All these required parameters are summarised in Table 1.

* Corresponding Author.

Email Address: kagan.koray@erzurum.edu.tr (K. K. Ayten)

<https://doi.org/10.21833/ijaas.2017.012.001>

2313-626X/© 2017 The Authors. Published by IASE.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

The origin of the coordinate system is located at the point where the rotary axis of actuator 2 is in Fig. 1. The maximum reachable point of the manipulator is approximately 0.6024 m from the actuator 2 base point. Link 2, link 3 and link 4 of the Katana manipulator (rotary joints 2, 3 and 4) are considered for the implementation of 3-link redundant manipulator. In this case, a system has more control inputs than required in order to control a specified desired motion.

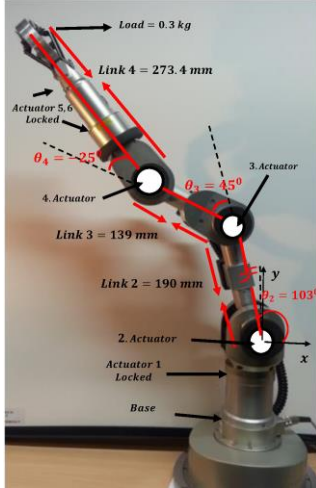


Fig. 1: Katana model with one redundancy in link-2

Table 1: Parameters for Katana 450

Parameters	Links		
	Link 2	Link 3	Link 4
Mass of Link (kg)	1.022	0.882	0.969
Length of Link (mm)	190	139	273.4
Distance from CoG to end link (mm)	95	35.3	163.4
Inertia of Link	0.0445	0.0445	0.0114
Friction coef. of Link (N.ms/rad)	1.8	1.5	0.39
Mass of gear (kg)	0.233	0.233	0.182
Gear ratio	371	371	100
Limits	Link 2	Link 3	Link 4
Absolute Angle (deg)	132	245	224
Relative Angle (deg)	+102/-30	+/-122.5	+/-112
Max velocity (deg/s)	72.52	73.53	136.8
Max acceleration (deg/s ²)	2321	2353	4378
Torque (N.m)	17	13	9

For the redundant scheme, the manipulator task consists of transporting a load mass of 0.3 kg from an initial point at $x_i = -0.3095$, $y_i = 0.4881$ m to a destination at $x_f = 0.1405$, $y_f = 0.4381$ m in Cartesian space as shown in Fig. 1. The total duration of motion is varied from 4 sec to 10 sec by increments of 2 sec.

3. The procedure of optimum trajectory planning

3.1. Formulation of the optimization problem and definition of the trajectory

Assumed that the n-links redundant manipulator task consists of transporting a load mass, an initial point to a destination in Cartesian space in Fig. 2. Cartesian coordinates of the centre of gravity and the relative angles of each link plus the Cartesian

coordinates of the load are selected as generalized coordinates, i.e. a total of $(3n+2)$ generalized coordinates for the n degrees of freedom system. DYSIM automatically develops $(2n+2)$ constraint equations and a constraint Jacobian matrix F. The last two relative angles are selected as dependant coordinates.

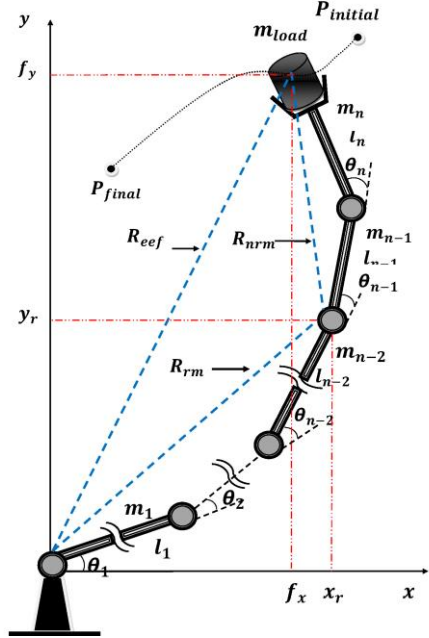


Fig. 2: Schematic view of redundant manipulator. Last two links are non-redundant

The n control input locations are selected to be the relative angles of all n links (i.e. the joint actuator torques). G represents the cost (objective) between initial and final postures, g_i is the required actuator torque to be applied at joint i, and T is the motion duration (Eq. 1).

$$G = \int_0^T (\sum_{i=1}^K g_i^2(t)) dt \quad (1)$$

3.2. Proposed methods

In conventional methods (such as the fmincon function in Matlab) the constraints equations are handled separately and the cost function is called regardless of whether the constraints are satisfied or not. In order to improve computational efficiency of the trajectory optimization algorithm, constraints can be handled within the cost function calculations. Another requirement for the motion defined by the optimization parameters to be realizable is that the distance between the end of the last redundant link (point at x_r, y_r) and the end-effector (point at f_x, f_y) must be less than or equal to $(l_{n-1} + l_n)$. This is equivalent to replacing the redundant links by a single virtual link R_{rm} as shown in Fig. 2. At each step of the movement, the position of the redundant/hyper-redundant robotic manipulators has to satisfy the following constraint, which can be formulated as Eq. 2. The following criterion must also satisfy the end-effectors reachable point Reef in Cartesian coordinates (Eq. 3).

$$R_{nrm} = \max \sqrt{((f_x - x_r)^2 + (f_y - x_y)^2)} \leq l_{n-1} + l_n \quad (2)$$

$$R_{eef} = \max \sqrt{((f_x)^2(t) + (f_y)^2(t))} \leq \sum_{i=1}^n l_i \quad (3)$$

4. Experimental implementation and results

The proposed methods were implemented in Simulink. Fig. 3 presents the comparison between the theoretically simulated optimized velocity profiles (reference) and the experimentally recorded optimized velocity profiles (actual) for links 2, 3 and 4, respectively. The velocities do not violate the

velocity constraints. As expected, experimental results of the velocities support the simulation results, and identical velocity profiles between the demand and actual velocity have been observed. As expected, the motion duration has a significant impact on the outcome of the cost function. It can be seen from the Table 2 that as the time of the motion is increased the energy consumption is increased. After optimization, the cost value is reduced remarkably, and the considerable improvement achieved approximately 56.57% in 4 seconds motion in theoretical study.

Table 2: Cost values

Time	Cost Values		
	Non-optimum	Optimum	Energy Saving %
Simulation (N^2m^2s)	(N^2m^2s)	(N^2m^2s)	(N^2m^2s)
Cost (4sec)	87	38	56.57
Cost (6sec)	128	62	51.48
Cost (8sec)	169	92	45.78
Cost (10sec)	211	122	42.21
Experimental (amp^2s)	Non-optimum	Optimum	Energy Saving %
Cost (4sec)	72	31	43.001
Cost (6sec)	99	60	39.22
Cost (8sec)	144	92	36.21
Cost (10sec)	176	119	32.54

The corresponding experimental cost values for varying durations of motions are also demonstrated in Table 2 and maximum energy reduction was observed to be approximately 43.001%, which corresponds to the duration of motion of 4 second profile in the experimental study.

4.1. Optimum trajectory planning for hyper-redundant manipulators

An 8-links hyper-redundant (with six DOFs redundancy) manipulator is introduced to verify the proposed methods for a large number of links. The manipulator task for this example is to move the load mass from an initial point $P_{initial}$ ($x_i=1.54$, $y_i=0.14$) m to a final point P_{final} ($x_f=0$, $y_f=-0.85$) m in Cartesian space coordinate as shown in Fig. 4.

The viscous friction effects of the joints are also included with a coefficient of friction '0.6' Ns/m and gears are also taken into account with all gear ratios $R=50$. The motion duration is specified as $T=2$ s. Mass centre of gravity of the links are in the middle of each link and the load mass is $m=0.3$ kg at the end of the last link. An 8-links hyper-redundant manipulator has eight identical links, and each link length is selected as 0.2 m.

For the each link, motor and link inertias were selected as 0.0001 kgm^2 and 0.0025 kgm^2 , respectively. All the links have identical mass, $m_l=0.5$ kg and each motor mass were selected as 0.2 kg. The corresponding randomly generated feasible non-optimum and optimum manipulative task are shown in temporal trajectory positions and also corresponding EEF's tracking trajectory as presented in Fig. 4. As is seen from the Fig. 4, the EEF tracking curve of the motion trajectory corresponding to non-optimum parameters has

almost a straight line trajectory apart from the beginning of the motion. This movement of the manipulator provides high a torque magnitude and sudden ascension on the non-optimized torque curves for all actuators between the duration of 1.3 and 1.4 seconds in the simulation as shown with dashes in Fig. 5.

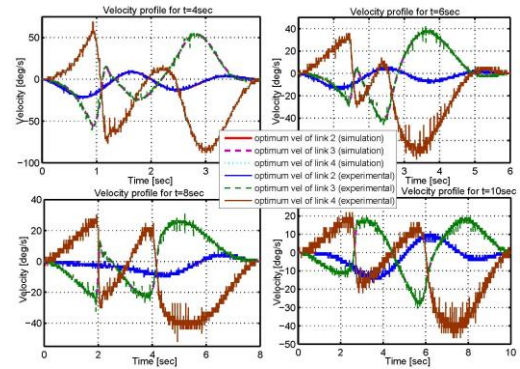


Fig. 3: Experimental comparison of speed with varying duration of motion

As is seen from the Fig. 4, the non-optimum cost curve increases rapidly and gives an initial cost value of $G=8084$. On the other hand, the EEF tracking curve of the motion trajectory corresponding to optimum parameters has a smooth EEF tracking profile as shown in Fig. 4b1.

This optimum motion of the manipulator provides admissible torque magnitude and avoids sudden increases in torque profile during the motion as shown in Fig. 5. After optimization, the cost function is reduced to $G=5232$, which corresponds to 35.3% reduction along the desired trajectory.

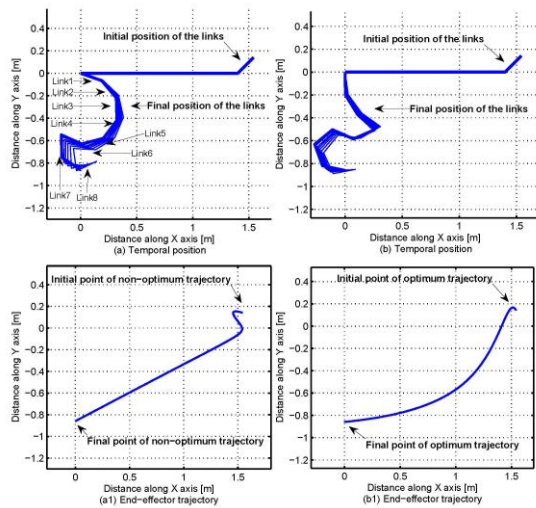


Fig. 4: (a), (a1) Motion trajectory corresponding to initial parameter, (b), (b1), Motion trajectory corresponding to optimum parameters

5. Conclusion

The simulation results make clear the effect of the proposed methods for the redundant/hyper-redundant robotic manipulators. The proposed methods not only achieve a reduction in the energy consumption for the hyper-redundant manipulators, but also have the ability of handling a large number of DOFs manipulators and constraints without any problems. A variety of constraints and different cost functions can easily be added to the proposed optimization procedure.

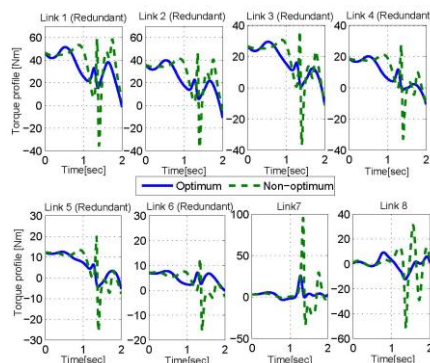


Fig. 5: Simulated planned trajectories of non-optimum and optimum torque profiles for hyper-redundant

References

- Assal SF, Watanabe K, and Izumi K (2006). Intelligent control for avoiding the joint limits of redundant planar manipulators. *Artificial Life and Robotics*, 10(2): 141-148.
- Ata AA and Myo TR (2005). Optimal point-to-point trajectory tracking of redundant manipulators using generalized pattern search. *International Journal of Advanced Robotic Systems*, 2(3): 239-244.
- Duarte FB, Machado JT, and Horvath L (1999). A trajectory planning algorithm for redundant manipulators. *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE'99)*, IEEE, 3: 1002-1007. <https://doi.org/10.1109/ISIE.1999.796759>
- Huo L and Baron L (2008). The joint-limits and singularity avoidance in robotic welding. *Industrial Robot: An International Journal*, 35(5): 456-464.
- Ma S and Watanabe M (2000). Minimum time pathtracking control of redundant manipulators. *IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, Takamatsu, Japan*, 1: 27-32. <https://doi.org/10.1109/IROS.2000.894577>
- Marey M and Chaumette F (2010). New strategies for avoiding robot joint limits: Application to visual servoing using a large projection operator. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE: 6222-6227. <https://doi.org/10.1109/IROS.2010.5653044>
- Patel RV, Shadpey F, Ranjbaran F, and Angeles J (2005). A collision-avoidance scheme for redundant manipulators: Theory and experiments. *Journal of Field Robotics*, 22(12): 737-757.
- Ping G, Wei B, Li X, and Luo X (2009). Real time obstacle avoidance for redundant robot. In the *International Conference on Mechatronics and Automation (ICMA'09)*, IEEE: 223-228. <https://doi.org/10.1109/ICMA.2009.5246286>
- Sahinkaya MN (2004). Inverse dynamic analysis of multiphysics systems. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 218(1): 13-26.
- Singla E, Tripathi S, Rakesh V, and Dasgupta B (2010). Dimensional synthesis of kinematically redundant serial manipulators for cluttered environments. *Robotics and Autonomous Systems*, 58(5): 585-595.
- Sullivan JCW and Pipe AG (1998). Path planning for redundant robot manipulators: A global optimization approach using evolutionary search. *IEEE International Conference on Systems, Man, and Cybernetics*, IEEE, 3: 2396-2400. <https://doi.org/10.1109/ICSMC.1998.725015>
- Yang J, Kim J, Pitarch EP, and Abdel-Malek K (2008). Optimal trajectory planning for redundant manipulators based on minimum jerk. In the *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, ASME, New York, USA: 1141-1150. <https://doi.org/10.1115/DETC2008-49268>