ANALELE UNIVERSITĂȚII
"EFTIMIE MURGU" REȘIȚA
ANUL XXVI, NR. 1, 2019, ISSN 1453 - 7397

Directional kinematics, first step in robotic movement

Ciprian Dragne

Before any robotic decision for moving to a new position, it's better to evaluate the actual robot position relative to the target, together with any trail obstacles and possible collisions with other object and special zones, or even self-collision during next steps. For this evaluation, the paper is developing a new method for the robot repositioning as a first step or including during the movement a forward or inverse kinematic algorithm with the aim to reduce steps until target location. This new method is named Directional Kinematics because it is addressed to the robot movement relative to a specific direction related to final target location.

***Keywords**: robotics, inverse and directional kinematics, movement strategy*

1. Introduction

If you have taken the first step in the right direction, you are already half way there at the right location. Probably you already have ever heard this expression. First step in finding a possibility to reach an object from the target location for humans or robots equipped with mobility is to turn with the *face* to the object and repositioning before any evaluation of the difficulties into grab and obstacles avoidance.

In case of the robots with arms, repositioning of the arm to reach the target location could be time expensive using only an inverse kinematic strategy. Repositioning of the arm links into appropriate position is a clearer strategy to reach to the target quicker and precisely. For this reason, this paper developed a new method that should be used before or during any forward or inverse kinematics strategy. This method represents an optimization step for a movement strategy of the robot to repositioning relative to target location.

1. Directional kinematics

Because inverse kinematic algorithms have not enough efficiency calculation for complicated solutions during real-time application, a new method to improve the efficiency of the inverse kinematics solution was proposed by introducing the directional concept. Unlike other methods, the proposed method establishes a pre-check of a real direction to move before any kinematics calculations. Furthermore, we adopt a new inverse kinematics algorithm, developing an improved sub-problem of the relative position between end-effector and the target location.

Definition: *Directional kinematics it's an additional kinematic step during Forward or Inverse kinematic algorithm in order to reach faster on the final target location.*

Directional kinematics consist into apply a new transformation matrix similar to transformation used into Jacobian Inverse kinematics technique, but when D-H (Denavit and Hartenberg) parameters are special evaluated [1]. These parameters are defined in this article as Directional Kinematic Parameters (DKP). Optimisation of DKP are made using a programming task defined in (1), where N_j is the number of robot arm joints used in DKP, P_j - the actual joint point location, P_{TG} - point location of the target, $f()$ - parameters function who depend of the deploying law of the joints over the robot working space.

$$\begin{aligned}&For\ i=1\ldots N_j\\&\min f(P_i-P_{TG})\Rightarrow\theta_i\ ;\\&Next\ i\end{aligned}\qquad(1)$$

Directional kinematics algorithm uses formulas for re-initializing of the D-H parameters oriented to the target position. This method can be used before or during of any Forward or Inverse kinematics algorithm. Directional parameters are calculated from minimizing of a functional with a special mathematical form. Effective improvement over the robot kinematics are evaluated for an Inverse Jacobian kinematic algorithm. DKP deploying functions could have vary polynomial representation. The D-H Parameters of the testing robot are represented in Figure 1, where the left is the real model and the right, the virtual 3D model. In plane D-H Parameters representation is displayed in Figure 2.

Depending on the configuration robot control loop could have implemented decisional evaluation algorithms based on target location and manipulability factors, and could select to act according to more than one movement strategy.

In the case of using multiple parameters for DKP, these parameters could be applied in vary order (not only in common order of robot link definition), depend on entire movement strategy and space restrictions into robot workspace (Figure 6).

Multiple strategies - total trail distance, energetically consumption or mechanical work (of the traveled trail) have been used for DKP deploying function evaluation criteria. Generating the robot movement trails considering the full robot dynamics allows to reason about multi-contact interactions on the complex surfaces. However, this results in high dimensional, non-convex and computationally complex algorithms, therefore limiting their applicability for real-time movement strategy and control it's highly recommended.

The robot positioning after using DKP in linear regression is presented in Fuigure 3, where left is the case of 2 parameters and right, the 4 parameters, respectivelly.

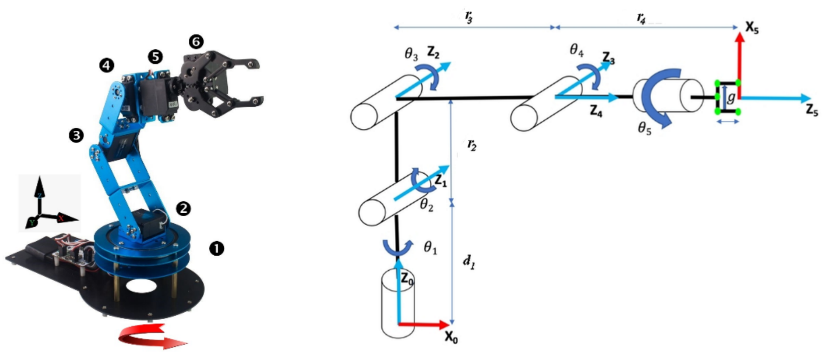


Figure 1. D-H Parameters of the testing robot
(left – real model; right - virtual 3D model).

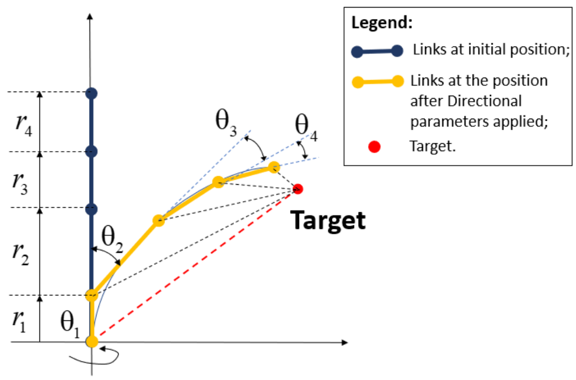


Figure 2. In plane D-H Parameters representation.

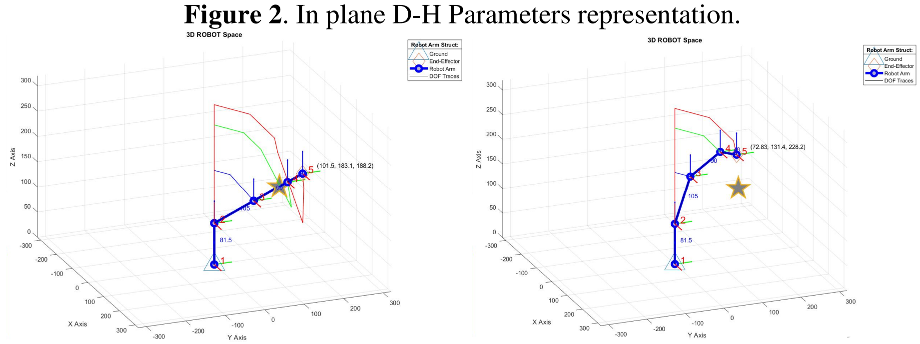


Figure 3. Robot positioning after using DKP in linear regression
(left – using 2 parameters; right – using 4 parameters)

3. Case studies

In the table 1 are presented case studies with target locations, directional parameters and initial and results data. For each case, the robot initial position is the rest (home) position presented in Figure 4-a.

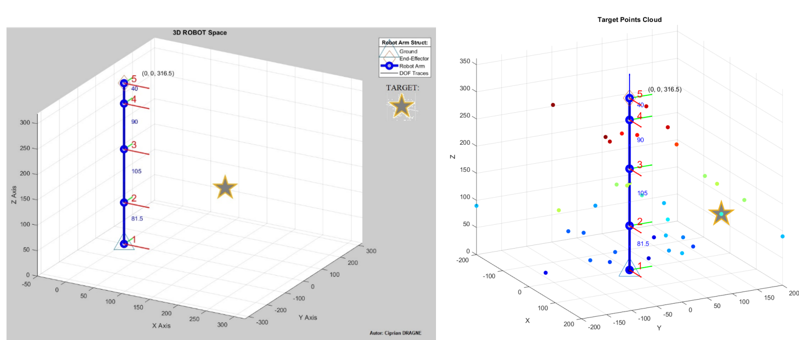
Table 1. Case studies details

Case	Initial data	Values [X, Y, Z] / [θ_1,θ_2,θ_3, etc]	Results after simulation		
			Steps wto. directional	Steps with directional	Improve [%]
1	Target no.1	[51,67, 236]	53	38	28
	DKP	[47, -21, 0]			

2	Target no.2	[78, -80, 272, 0]	54	40	26
	DKP	[-46, -30, 0, 0]			
3	Target no.3	[78, -80, 50, 0]	48	42	13
	DKP	[-45, -90, 0, 0]			
4	Target no.4	[78, -80, 50, 0]	48	38	21
	DKP	[50, -30, 0, 0]			
5	Target no.4	[78, -80, 50, 0]	48	33	31
	DKP	[50, -30,-90,-90]			
6	Target no.5	[75,135,160]	49	34	30
	DKP	[61, 32, 43, 0]			
7	Target no.6	[25, -180,30, 0]	51	42	17
	DKP	[82, 53, 36, 0]			
8	Target no.7	[120,120,120, 0]	59	33	44
	DKP	[45, 39, 56, 0]			
9	Target no.8	[150,150,150,0]	46	38	17
	DKP	[45, 36, 30, 0]			
10	Target no.9	[0.1,0.1,160, 0]	51	42	17
	DKP	[45, 0, 60, 0]			

For the cases [1 ÷ 3, 6 ÷ 10] different target locations are used depending on the side relative to initial position of the robot (home position). For cases 3 ÷ 5 was used the same target location, only with different D-H parameters used for directional kinematic algorithm.

In table 1 are presented only a part from all target points used for testing. All target locations are presented in Figure 4.b.



a. b.

Figure 4. Robot Arm 3D space

a. Home position, b. Cloud of all target points used for testing

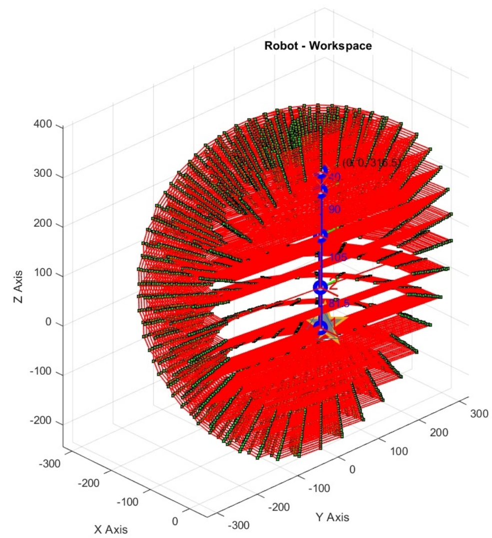


Figure 5. Robot working space in half-representation

4. Robot movement strategy

Most of the kinematic movements during optimization positioning of end-effector to a certain target appear to be into areas near target location, not far away from it. This can be understood by studying the end-effector coordinates updating during the kinematic movements using inverse kinematics or inverse kinematics with DKP algorithm.

For this reason, a robot strategy should be divided in two different type of controlled movement: automatic control for fine and precisely adjustments and global control for positioning into passive phases of surgery.

A robot acting using motors must be designed, sensed, actuated and controlled by programming tools. Such devices should have different sensitivities depend on strategy of robot movement.

A strategy using kinematic algorithm that can use a more free selection of DOF that should be manipulated by changing DOF order in action, changing actuators sensitivity, decoupling of translational and rotational motions or other kinematic measure to take of advantages from gravity compensation should be put into designing plan of any robot even from beginning of its concept phase.

An optimal path planning algorithm should take into consideration vary optimization strategy based on: precision in movements, mobility near target locations, the length of the trails, DOF involved, energy consumption and work load necessary for route.

A controlled movement strategy of the robot should be correlated with information from sub-systems for real-check positioning for a complete validation of the robot position. These sub-systems should include motors with joint encoders for accurate motion and a navigation sub-system used to locate the position of the patient during real surgery time.

5. Collision detection avoidance

In robotics for satisfying some control objective like the precise end-effector positioning, the precision movements into certain directions or controlled action of gripper devices, it is necessary to implement the programming routines for obstacle avoidance, non-intersection or non-collision and position constraints.

Path planning and avoidance techniques for controlled movement strategy of the robots who action over driverless platform required a robust obstacle detection module [3]. Because of the large amount of input data related to the interference environment which you need to implement, the collisions detection between objects in a virtual environment is rather tedious. This makes the implementation of a collision difficult and depend on number of obstacles that are needed to be avoid. The complexity of the geometry exponentially expands the search interactions of the possible interference areas and thus of the validation time of the decision whether or not to interfere. The routines should be implemented in a clear manner and each section validated by results data analysis and visually inspection check.

6. Conclusions

A robot strategy should be divided in two different types of controlled movement: automatic control for fine and precisely adjustments and global control for positioning into passive phases of surgery. Both type of robot movements should be correlated with informational results from a navigation system that should check real positioning of the robot.

Depending on the configuration of the robot and the control loop programming and implemented routines, the robot control system could have implemented by decisional evaluation algorithms based on the target location, path planning, collision avoidance and detection, DOF selection, mobility near target location, energy consumption and even decisional factors based on the gravity compensation during movement.

A robot strategy should take into the consideration the number of objects that should be avoided during movement until target location.

Acknowledgement. This work was supported by a grant of the Romanian ministry of Research and Innovation, CCCDI–UEFISCDI, project number PN-III-P1-1.2-PCCDI-2017-0221/59PCCDI/2018 (IMPROVE), within PNCDI III.

References

[1] Dragne C., *Robotics Design -Inverse and Directional Kinematics,* SISOM Conference 2019
[2] Richard P., *Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators,* MIT Press, Cambridge, MA, 1981.
[3] Becker M., Dantas C.M., Macedo W.P., *Obstacle Avoidance Procedure for Mobile Robots*, In: Paulo Eigi Miyagi, Oswaldo Horikawa, Emilia Villani (Org.) ABCM Symposium Series in Mechatronics, São Paulo (Ed) SP: ABCM, 2, 2006, pp. 250-257.
[4] McCarthy J.M., *Introduction to Theoretical Kinematics*, MIT Press, Cambridge, MA, 1990.
[5] Denavit J.D., Richard Scheunemann H., A kinematic notation for lower-pair mechanisms based on matrice*, Trans ASME J. Appl. Mech.*, 23, 1955, pp. 215–221.
[6] Hartenberg R.S., Denavit J., *Kinematic synthesis of linkages*, McGraw-Hill series in mechanical engineering, New York, McGraw-Hill, p. 435, 1965.
[7] Richard P., *Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators,* Cambridge, MA: MIT Press, 1981.
[8] Ericson C., *Real-Time Collision Detection,* The Morgan Kaufmann Series in Interactive 3-D Technology, CRC Press, Dec. 22, 2004.
[9] Munteanu L., Dragne C, Rugina C., Dumitriu D., Ionescu M., Chiroiu V., *A review on the optimum design criteria in robotics,* SISOM Conference, 2019.

Address:

- Eng. Ciprian Dragne, Assistant Research Engineer at the Institute of Solid Mechanics, Romanian Academy, C-tin Mille, no.15, Bucharest, ciprian_dragne@yahoo.com