Conférence Internationale en Automatique & Traitement de Signal (ATS-2017)
Proceedings of Engineering and Technology – PET
Vol.22 pp.84-89

Grasping and manipulating object in the environment with and without obstacles

Chaabani Ali[#1], Mohamed Sahbi Bellamine[#2], Moncef Gasmi[#3]

[#]*Computer Laboratory for Industrial Systems, National Institute of Applied Sciences and Technology, University of Carthage, TUNISIA*
[1]*chaabani.ali@gmail.com*
[2]`aroussia@insat.rnu.tn`
[3]`mcfgsm@yahoo.fr`

***Abstract*—This paper merges path and manipulation planning techniques to control the PR2 (Personal Robot 2) in the environment with and without obstacles. In much previous work on grasping, the object being grasped is assumed to be the only object in the environment and it is near the robot. In this paper, the object is surrounded by obstacles. Hence the PR2 needs to avoid obstacles and navigate to the target object and grasp it. We introduce an expansion of the RRT planning algorithm to drive the PR2 robot which is equipped with 7-DOF torque controlled arms and manipulate each joint of its robotic arm plus its end effector. We present experiments in simulation and on the PR2 robot.**

Keywords— PR2; trajectory; obstacles; navigate; planning;

I. Introduction

This Mobile robots have many application areas thanks to their high workability. Researchers have exhibited an interest in mobile robots. So, it's important to have mobile robots able to avoid obstacles and grasp the objects. The problem of avoiding barrier and capturing targets using the volatile manipulator is a compound task, [1]. The issue of seizure a thing in the existence of holdbacks with an automatic manipulator was exposed by a lot of studies. Many studies have treated the handling navigation problem for a manipulator to avert accidents with obstacles [2]. The robots catching capability of a given object is substantial and the motion planning of prevailing the manipulator to seize the items without clash is thoughtful for an expanded time. Rapidly-exploring trees (RRTs) is one of the sampling-based of the planning algorithm [3] and those of the Probability Roadmap (PRM) [4] are public in new years for the reason that of their capability of fast discovering the connection of large-dimensional configured spaces. Nevertheless, the manipulator aspire over the interaction of the human-robot is not steady but dynamic. The movable robot had become an essential research topics. Most published research in control of movable manipulator behold its dynamics and kinematics [5].
The object localization is a very complex process in traditional algorithms. Moreover, the employ of the predictive algorithms may cause different issues in the complexity of the algorithms that are based on many calculation and estimation. In this effort, we propose an approach allowing to control the PR2. The goal of our approach is to evade obstacles and grasp the object. We propose an algorithm, which is a stretching of RRT-JT, to generate the shortest time path to apprehend the target, while avoiding obstacles. Finally, LQR method is employed for tracking purposes.

We present how to clutch a target with a fixed manipulator: the object is in the sphere which the diameter is the arm's length, therefore where the object is far, the robot can't grasp it anymore. To avoid that situation we utilize a mobile robot which is an assembly of two main components: a mobile base and PR2 robot.

The article is ordered as following: The related work has been detailed in part II. Then, in part III, we confer the system overview. The part that followed describes the PR2 robot. In Section V, we attend the robot motion control. The part VI contains the optimal linear control. Finally, in section VII a several results are given, then the conclusions.

II. Related work

Each in previous work, researchers have divided the matter of moveable manipulation planning into four master techniques: navigation planning of the robot's base, path planning for a robot's arm, grasping and frameworks for generic manipulation planning. There are many researchers who are concerned in object discovery and tracking using mobile robot under unknown dynamic environments [6] [7]. [8], oncoming the problem of the planning of motion is focused on positioning the end effector in the pre-configured locations, counted utilising the inverse kinematics(IK) used to some primary specimens possessed from the aim region. Therefore, these locations are collected as an objective for a planner that is randomized as well as RRT and BiRRT[9]. The resulting solution provided by this way is deficient because of the miss deem probabilistic aspect, and the planner is required to use the prechosen numbers in the goal regions. Many people treat the manipulation of PR2 robot[10,11,12]

III. System overview

We utilize the OpenRAVE which is mostly used for storing the environment symbol of the manipulators, sensor situation, other scene patterns… We utilize the OpenRAVE software to command the PR2 robot with mobile base. Our objective is to follow and pinch the object while evading barriers in the

ISSN 2356-5608

environments. The approach in our paper uses an RRT-JT to determinate the target's position and to calculate the kinematics of the movable robot while avoiding obstacles.

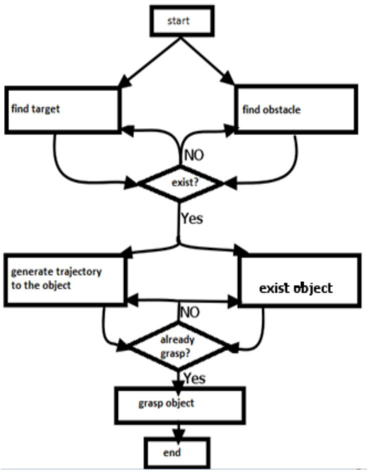


Fig 1 Flow chart: local motion planner.

The flow chart is in Fig. 1. Primary, we want to locate the goal and the obstacle, if it exist, we command the mobile robot to displace to the goal while avoiding the existing obstacle : trajectory generation and object simulation should be done in parallel. At last, the robot seizes the target.

IV. The PR2 robot

The PR2 service robot has seven joints, which adds a redundancy to the kinematics.
This issue will be discussed in further details below, but for now we assume that the value of the first joint j0 is set by the user

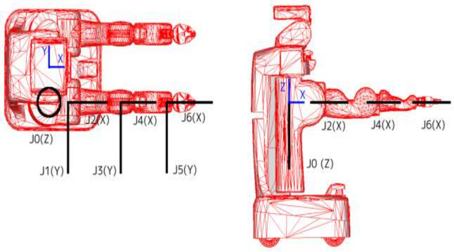


Fig. 2 The labeled joints (black) of the PR2 robot's right arm.(refe)

Fig.2. displays the full 7-DoF of the PR2 robot's right arm in the first situation.

An affirmative joint movement is on the right hand rule, for every one axis. To fix the transformation between the axes K and K-1, we employ the following equation of homogeneous conversion in (1).

(1) D-H generalized transform matrix

$$ {}^{k-1}T_k \begin{bmatrix} \cos\theta k & -\sin\theta k\cos\alpha k & \sin\theta k\sin\alpha k & \alpha k\cos\theta k \\ \sin\theta k & \cos\theta k\cos\alpha k & -\cos\theta k\sin\alpha k & \alpha k\sin\theta k \\ 0 & \sin\alpha k & \cos\alpha k & dk \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

- ak-1 = distance from Zk-1 to Zk measured over Xk-1
- dk = distance from Xk-1 to Xk considered along Zk
- αk-1 =angle between Zk-1 to Zk was roughly Xk-1
- θk =angle amidst Xk-1 and Xk was approximately Zk

As with the prior example, we describe accurately the frame for our particular end effector. By multiplying all of the transforms up to and enclosing the final frame, we settle officially the forward kinematics for any frame on the robot. The fundamental chain of transformations that compute the end effector link frame is defined by:

$$ {}^{0}T_{Tool} = {}^{0}T_1\, {}^{1}T_2\, {}^{2}T_3\, {}^{3}T_4\, {}^{4}T_5\, {}^{5}T_6\, {}^{6}T_7\, {}^{7}T_{Tool} \quad (2) $$

The conversion equations used to bring up the manipulator's joints pending the interval between the object and the end effector nearly equivalent to zero. We show how to solve a subset of the joints of the PR2 for translation. We treat the wrist position where the last three joints intersect as the translation target. Once the assignment of the contact is accomplished, the grippers grapple the object.

V. Robot motion control

A. Robot dynamics

The Lagrange equations were utilized by the dynamic simulation to acquire the angular acceleration from the torque of every joint. Foremost, we computed the body Jacobian of every joint J_i corresponding to M_i, where M_i is the ith joint's inertia matrix. So the manipulator inertia matrix $M(\theta)$ can be calculated as

$$ M(\theta) = \sum_{i=1}^{n} J_i^T(\theta) M_i J_i(\theta) \quad (3) $$

Also calculate potential part,

$$ M(\theta) = \sum_{i=1}^{n} J_i^T(\theta) M_i J_i(\theta) \quad (4) $$

Second, we calculated the torques of all joints using Lagrange equation, which is,

$$ \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \left(\frac{\partial L}{\partial \theta_i}\right) = U_i \qquad (i = 1 \ldots 7) \quad (5) $$

where represents the torque of the joint and,

$$L(\theta,\dot{\theta}) = \frac{1}{2}\ \dot{\theta}^{T} M(\theta)\dot{\theta} - P(\theta) \quad (6)$$

After expanding the components of the equation, the equation is as follows:

$$M(\theta)\ddot{\theta} + H(\theta,\dot{\theta}) + G(\theta) = U(t) \qquad (7)$$

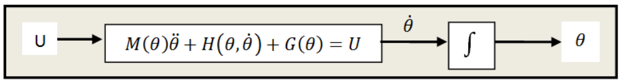


Fig. 3 Block graph of the open loop system

where $H(\theta,\dot{\theta}) = B\ (\theta).\dot{\theta}^2$ is the Coriolis and centrifugal force and $G(\theta)$ is the gravity term and

$$\theta = [\ \theta_1\ \theta_2\ \theta_3\ \theta_4\ \theta_5\ \theta_6\ \theta_7]\ and\ \dot{\theta} = \frac{d\theta}{dt}\ ,\ \ddot{\theta} = \frac{d^2\theta}{dt^2}$$

The potential energy P_i of an element C_i of mass m_i of the robotic system expressed to the base R_0 is written as:

$$P_i = -m_i g\, {}^0\bar{r}_i = -m_i g({}^0T_i\, {}^i\bar{r}_i)$$

The sum potential energy P of our arm manipulator:

$$P = \sum_{i=1}^{7} P_i = \sum_{i=1}^{7} -m_i g({}^0T_i\, {}^i\bar{r}_i)$$

$${}^i r_i = [x_i \quad y_i \quad z_i \quad 1]^T$$

Where ${}^i\boldsymbol{r}_i$(the location vector of the material point $\boldsymbol{m}_i$) and the gravity vector:

$$g = [0 \quad 0 \quad -9.8032 \quad 0]^T$$

Given that the mobile base is strolling in a horizontal plane its potential energy $P = 0$.

Mappings between the joint coordinates and the robot end-effector coordinates X_r are given as

$$X_r = W(\theta) \qquad (8)$$

$$\dot{X}_r = J(\theta)\dot{\theta} \qquad (9)$$

$$\ddot{X}_r = \dot{J}(\theta)\dot{\theta} + J(\theta)\ddot{\theta} \qquad (10)$$

Where $W(\theta)$ describes the onward kinematic relation for the end-effector $J(\theta)$ and is the end-effector Jacobian matrix. By substituting (8)–(10) into (7), one can acquire the robot's dynamic equation in cartesian space

$$MJ^{-1}\{\ddot{X}_r - \dot{J}\,J^{-1}\dot{X}_r\} + HJ^{-1}\dot{X}_r + G = U \qquad (11)$$

By rearranging the terms, one can obtain the robot's dynamic equation of motion as

$$MJ^{-1}\ddot{X}_r + \{H - M\,J^{-1}\dot{J}\}\,J^{-1}\dot{X}_r + G = U \qquad (12)$$

B. Using the Jacobian

We are occupied in reckoning an extension in configuration space from q∈ Q towards xg∈ X, when the preferred end effector aim xg and the robot arm configuration q are given, where X is the positions of the end of the robot arm in R^3. In spite of that the mapping from Q to X is frequently nonlinear and hence expensive to deduct, its derivative the Jacobian, is a linear map from the tangent space of Q to that of X, that can be calculated easily (J $\dot{q}=\dot{x}$, where x∈ X is the end effector location corresponding to q). Ideally, to urge the end effector to a wished configuration xg we could compute the error e(t)=(xg −x) and hurry a controller of the shape $\dot{q}= KJ^{-1}e$, K is a positive gain. This uncomplicated controller is able to acquire the target with no considering of any feasible barriers or articulation limits. Though this rotate inside a compound controller, where the converse of the Jacobian should be done at every moment step. To escape this expensive approach, we use alternatively the transpose of the Jacobian and the control rule collapse into the form of $\dot{q}= KJ^{T}e$. The controller expels the spacious overhead of computing the inverse by using the easy-to-compute Jacobian instead. The instantaneous action of the end effector is given via $\dot{x} = J\dot{q} = J(KJ^{T}e)$. The inner creation of this instantaneous motion with the fault vector is specified by $e^{T}\dot{x} = ke^{T}JJ^{T}e \geq 0$. As this is forever positive, under our assumptions with obstacles, we may ensure that the controller will be clever to make onward progress towards the target.

C. Robot's velocity

From the first position $a(x_1,y_1,z_1)$ to the second position $b(x_2,y_2,z_2)$ which $x_2 = x_1 + \Delta_x$, $y_2 = y_1 + \Delta_y$ and $z_2 = z_1 + \Delta_z$ and $t_{a\to b} = t_{sleep} + \varepsilon$ which t_{sleep} is the time to rest en a and ε is the time from a to b.

$\mathbf{V_x} = \frac{\boldsymbol{\Delta_x}}{\mathbf{t_{a\to b}}}$, $\mathbf{V_y} = \frac{\boldsymbol{\Delta_y}}{\mathbf{t_{a\to b}}}$ and $\mathbf{V_z} = \frac{\boldsymbol{\Delta_z}}{\mathbf{t_{a\to b}}}$ so the robot's velocity

$$\mathbf{V_T} = \mathbf{V_x i} + \mathbf{V_y j} + \mathbf{V_z z} \quad (13)$$

VI. Optimal linear control

The synthesized Control U moves the system from an initial position to an equilibrium θ_{eq} (constant characterized by acceleration and velocity joint equal zero $\dot{\theta}_{eq} = \ddot{\theta}_{eq} = 0$).

This command is written like this: U = U_{eq}+ V

A. *Case study of balance*

When the robot reaches its equilibrium was then: $G(\theta_{eq}) = U_{eq}$

Where U_{eq} is the command required to preserve the robot to the position of static equilibrium θ_{eq} .

- ✓ **Linearization**

This linear order, as the name suggests, is bottomed on the linearization of the equations of dynamic of the robotic system.

To do this, the following definitions are provided:

-ϕ is the variation of θ relative to $\phi = \theta - \theta_{eq}$

- Taylor expansion to the first seek of a function f (θ) near θ_{eq} is as follows*:*

$$f(\theta) = f(\theta_{eq} + \phi) = f(\theta_{eq}) + \left(\frac{\partial f}{\partial \theta}\right)_{\theta_{eq}} \phi \quad with\ |\phi| \ll 1$$

By applying the development to the robotic system and employing the equations defined above, the dynamic equation of the robot becomes:

$$\left(J(\theta_{eq}) + \left(\frac{\partial J}{\partial \theta}\right)_{\theta_{eq}} \phi\right)\ddot{\phi} + \left(B(\theta_{eq}) + \left(\frac{\partial B}{\partial \theta}\right)_{\theta_{eq}} \phi\right)\dot{\phi}^2 + \left(G(\theta_{eq}) + \left(\frac{\partial G}{\partial \theta}\right)_{\theta_{eq}} \phi\right) = U_{eq} + V$$

After simplification and $G(\theta_{eq}) = U_{eq}$, the equation of the linearized dynamics:

$$V = J(\theta_{eq})\ddot{\phi} + \left(\frac{\partial G}{\partial \theta}\right)_{\theta_{eq}} \phi$$

- ✓ **state representation:**

The linear system manifested by the over equation is a matrix representation translates following state: $\dot{X} = AX + BV$ where $X = \begin{bmatrix}\phi \\ \dot{\phi}\end{bmatrix}$ and $\dot{X} = \begin{bmatrix}\dot{\phi} \\ \ddot{\phi}\end{bmatrix}$

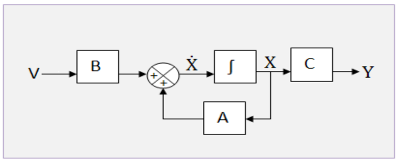


Fig. 4 Block diagram of the linearized System

B. *Calculation of the optimal control:*

The optimality criterion to minimized, put under linear quadratic form, is:

$$J(X,t) = \int_{t}^{tf} (X^T Q X + V^T R V)\, d\tau$$

Where Q and R matrices are positive definite and the optimal control problem is called: problem LQR (Linear Quadratic Regulator).
To locate a solution to this problem LQR, suppose that the minimum cost standard is also quadratic form as:

$$J^*(X,t) = X^T S X$$

The S matrix is symmetric, positive and is a function of time, as: $\begin{cases} S(t) = S(t)^T \\ \dfrac{\partial J^*}{\partial t} = X^T \dot{S} X \\ \dfrac{\partial}{\partial V}\left(\dfrac{\partial J^*}{\partial t}\right) = 0 \end{cases}$

The aim therefore is to search the optimal control *V** that reduces the cost criterion:

$$V^* = -R^{-1} B^T S X$$

Where S, positive definite, is the sole solution of the next Riccati algebraic equation:

$$Q + SA + A^T S - SBR^{-1} B^T S = 0$$

C. *Algorithm:*

1- move the robot base to target: *SetActiveDOFs()* and *MoveActiveJoints()*

2- move the arm to the target: *MoveToHandPosition()*

3- close fingers until collision: *CloseFingers()*

4- move the arm with the target back to the initial position: *MoveManipulator()*

5- move the robot to another location:

```
robot.GetTransform()
  robot.SetActiveDOFValues(goal)
  incollision = env.CheckCollision(robot)
    if incollision:
        print 'goal in collision!!'
 basemanip.MoveActiveJoints()
```

6- move the arm to the designated position on another table to place the target down:
MoveToHandPosition()
taskprob.ReleaseFingers()

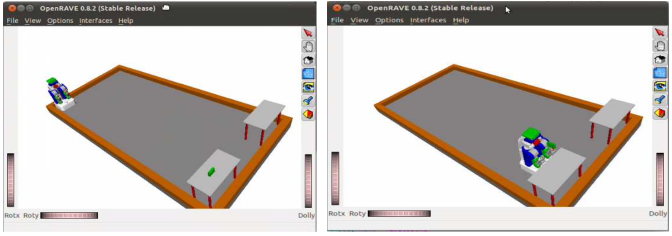
7- move manipulator to initial position: *MoveManipulator()* and *CloseFingers()*

VII. Results and analysis

To demonstrate the effectiveness of our RRT based algorithm on our PR2 robot, we ran several experiments. We utilize the OpenRAVE simulator to drive the PR2 robot and control its arms and its grippers for grasping the object. The objective is to hold and manipulate a model object in the environment with and without obstacles. In order to stably follow and grip the target, the PR2 avoids collision in its way to attain the object than it closes its fingers. The gripper would budge right or left with the shorter distance. The chore is to shift an object from some assumed beginning place to another given aim place. The matter can be splintered down into four sub-problems: 1) move the PR2 to the object, 2)move the gripper from its beginning configuration to a configuration where it is approximate the target, 3) seize the object, 4) move the robot (holding the object) to some configuration which puts the object into its goal configuration.

A. Grasping Objects in the Environment without Obstacles

As shown in Fig. 5, the OpenRave was used to simulate the robot. These are the default positions used : of the PR2 were (-3.4m, -1.4m, 0.05m) and those of the object were (3.5m, -1.3m, 0.74m).



(a) Initial position T=0s (b) Final position T=2.01s

Fig.5 As an example of a successful case of confining an object.

To retrieve the object saved in another table , the robot has to find the shortest path as indicated in Fig.5

TABLE I. ENVIRONMENT WITHOUT OBSTACLES

	Tgrasp(s)	**Tend(s)**
Trial1	2.00	5.22
Trial2	2.02	5.20
Trial3	2.01	5.25
Trial4	2.01	5.17
Trial5	1.98	4.83
Trial6	1.99	4.90

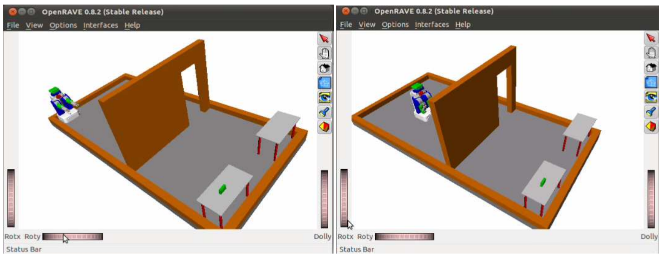
As presented in the Table I , two variables (times) are mentioned :

-Tgrasp : the outcomes of the time provided to the grip

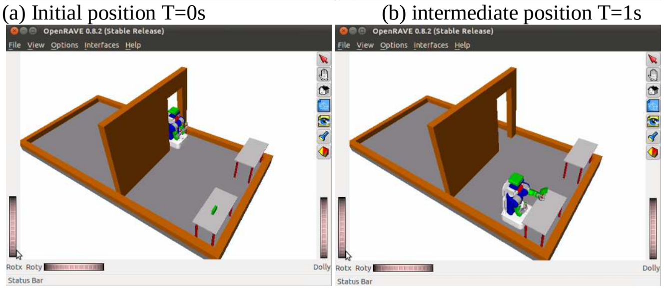
-Tend : the time to shift the object to the preferred position

The major goal of PR2 is to optimize its trajectory by searching the shortest path , so it is always reducing the time that ensures an elevated success rate of grasp

B. Tending to grasp Grasping Objects in the Environment with Obstacles



(a) Initial position T=0s (b) intermediate position T=1s



(c) intermediate position T=3s (d) Final position T=4.02s

Fig. 6 The robot turn away barrier, and, touch the object

The PR2 stays away from the obstacles by which one is surrounded, and, it grasps the target, then, it puts it in the required position, and, opens its fingers. Environment with obstacles.

TABLE II. REPRESENT AN ENVIRONMENT WITH OBSTACLES

	Tgrasp(s)	Tend(s)
Trial1	4.02	6.95
Trial2	4.07	6.90
Trial3	3.99	6.87
Trial4	4.01	6.84
Trial5	3.97	6.82
Trial6	4.00	6.85

The results of the time when the PR2 is avoiding obstacles are illustrated in the table II.

It is noticeable that time is decreasing between the first and the last test because the robot 's goal is to optimize a distance-based cost function and also the time to seize between the two configurations.

To reach its goal, the robot spends much time because of the obstacle which is idling its velocity and complicating the grasping.

After many trials, PR2 was performing its work with a good result confirming its competence to successfully chase the object using the tracking information. It should be noted that the time spent was very reasonable.

Conclusions

In this manuscript, we studied the grasping capability of an object in real time while trying to shun the obstacles with the PR2. The PR2 is the combination of a 7-DoF robotic arm having a gripper and the mobile base in which we require the RRT algorithm. We have performed an empirical evaluation of grasping objects, while shunning collisions. Actually, this application permits to prevail the difficulty of the opposite kinematics using the Jacobian's element as a conversion tool from a configuration space to workspace. Our algorithm is able to choose the optimal grasp and positions for the mobile base. We fixed forth the clutching monitor of the object. The intended algorithm successfully holding the object in a rational time putting it in the aim station and the robots tries to optimize the distance between two configurations and the time to seize the object. In the presence of the obstacles in the environment the path becomes more complex.

References

[1] E. G.Papadopoulos . and J. Poulakakis, "Planning and Obstacle Avoidance for Mobile Robots,"Proceedings of the IEEE International Conference on Robotics and Automation, Seoul, Korea, 2001.

[2] B.Cohen, M.Phillips, S.Dhonde and M.Likhachev, "Planning Single-Arm Manipulations with N-Arm Robots, "Eighth International Symposium on Combinatorial Search, 2015.

[3] J. Kuffner, Jr. and S.M. LaValle , "RRT-connect: An efficient approach to single-query path planning", in Proc, IEEE Int. Conf. Robotics and Automation, 2000, pp. 995-100.

[4] L. Kavraki, P. Svestka, J. C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration space," IEEE Trans. Robotics and Automation, vol. 12, no.4, 1996, 566-580.

[5] R. Siegwart, M.Chli and M.Rufli, "Autonomous Mobile Robots," Spring, 2016.

[6] M. Almasri, K.Elleithy and A. Alajlan, "Sensor Fusion Based Model for Collision Free Mobile Robot Navigation, " Sensors, 2016, 16- 24.

[7] Z.A. Ali, D. Wang, M. Safwan, W. Jiang and M.Shafiq, "Trajectory Tracking of a Nonholonomic Wheeleed Mobile Robot Using Hybrid Controller," International Journal of Modeling and Optimization, Vol. 6, No. 3, June 2016.

[8] J. K. M. Stilman, J.-U. Schamburek,T. Asfour, "Manipulation planning among movable obstacles, " in IROS, 2007.

[9] S.M. LaValle, " Planning algorithms," Cambridge, U.K.: Cambridge University Press, available at http://planning.cs.uiuc.edu/(2006)(Accessed on 17/04/2016).

[10] S.Levine, C.Finn,"End-to-End Training of Deep Visuomotor Policies",Journal of Machine Learning Research 17, 2016, 1-40.

[11] X. shaukat, M.Williams "Human pointing as a robot directive "ACM/IEEE international conference on Human-robot interaction,2013.

[12] C.Amato, G.Konidaris, Ariel Anders and G. Cruz, "Policy Search for Multi-Robot Coordination under Uncertainty", 2016.

ISSN 2356-5608