A RRT-based collision-free and occlusion-free path planning method for a 7DOF manipulator

Conference Paper · August 2014

DOI: 10.1109/ICMA.2014.6885837

6 authors, including:

Yeoun-Jae Kim
Asan Medical Center
27 PUBLICATIONS 81 CITATIONS

Jeonghyeon Wang
Korea Advanced Institute of Science and Technology
13 PUBLICATIONS 20 CITATIONS

Joon-Yong Lee
Pacific Northwest National Laboratory
62 PUBLICATIONS 302 CITATIONS

Jeong-Jung Kim
Korea Institute of Machinery and Materials
32 PUBLICATIONS 180 CITATIONS

A RRT-Based Collision-Free and Occlusion-Free Path Planning Method for a 7DOF Manipulator

Yeoun-Jae Kim
Robotics Program
KAIST
Daejeon, Korea
Email: lethkim@empas.com

So-Yeon Park
Department of Electrical Engineering
KAIST
Daejeon, Korea
Email: arashi@kaist.ac.kr

Jeong-Jung Kim
Department of Electrical Engineering
KAIST
Daejeon, Korea
Email: rightcore@kaist.ac.kr

Jeong-Hyeon Wang
Robotics Program
KAIST
Daejeon, Korea
Email: leoneisjh@naver.com

Joon-Yong Lee
Department of Genetics Development and Cell Biology
Iowa state university
Ames, USA
Email: junyoni@gmail.com

Ju-Jang Lee
Department of Electrical Engineering
KAIST
Daejeon, Korea
Email: jjlee@ee.kaist.ac.kr

***Abstract*—In this paper, we present a path planning algorithm of a 7DOF manipulator with RRT(Rapidly-exploring Random Trees). The method presents a occlusion-free and collision-free path planning algorithm. First, we present a basics of RRT algorithm which is a basic searching method for a path planning of this paper. Second, the implementation of a occlusion-free path constraints is presented. Third, the collision-avoiding path planning constraints is presented. Fourth, the integration of RRT with these two constraints is shown. The proposed method is verified by a numerical simulation with Matlab and v-rep robotics simulation tool.**

***Index Terms*—7DOF manipulator, Path planning, Occlusion, Collision, Simulation.**

I. INTRODUCTION

Until recently, many studies on humanoid robots are addressed by a wide variety of researchers. Asimo[1], Hubo[2] and Petman is just a glimpse of such a humanoid robot with which researchers made a research. The major research areas of humanoid robots are divided by three categories. One is the intelligence of a humanoid robot. This research area is about the analyzing and decision making algorithm of the humanoid robot. Machine learning and data mining is major subjects to the intelligence of a humanoid robot. Another is the moveability of a humanoid robot. Moveability means the moving ability of the parts that comprises a humanoid robot. Walking, hopping, running and arm manipulation are such a research area. The other is the sensing of a external environment of a humanoid robot. Mainly vision with camera and contact sensors for hand manipulation are the research area regarding the sensing of an external environment of a humanoid robot.

Of all the research areas, the manipulation of arm of a humanoid robot is important because a humanoid robot must move its arm to reach the target object for it to manipulate. Basically, a humanoid robot is made to resemble a human-being, and most of the time, a human-being moves and manipulates its arm and hand to perform a meaningful work. However, while a human-being can move its arm to the target object with ease, mimicking this kind of functionality with a humanoid robot is not an easy task. Moreover, a humanoid robot must evade collision of its arm with other obstacles during the movement of its arm to the target object. This functionality is especially important because if the obstacles are human-being, the robot must halt the moving of its arm or at least dodge the human-being not to injure him(or her). The occlusion is another issue for a humanoid robot to overcome to manipulate its arm. A humanoid robot depends on vision by cameras which is attached and integrated to a humanoid robot and if the occlusion happened during the manipulation of the arm of a humanoid robot, some functionality by a humanoid robot which depends on the vision can be interrupted.

Regarding the path-planning of the manipulator, the use of visual sevoing technique is firstly addressed [3][4]. The basic use of this visual servoing technique is to linearizing the system by using 1st and 2nd derivatives. And some constraints are added to the visual servoing technique for the path planning of the manipulator [5][6]. Baumann et. al.[7] generated a occlusion-free path planning with a PRM(Probabilistic RoadMap). Leonard et. al. made a collision-free and occlusion-free paths with a PRM for industrial manipulator [8].

In this paper, we presents a occlusion-free and collision-free path planning algorithm of 7DOF manipulator of a humanoid robot with RRT. The above works used a probabilistic roadmap to generate a path planning. However, RRT is a powerful tool for the path planning application area for its convergence and speed of the algorithm. Thus, we make use of RRT to generate a occlusion-free and collision-free path planning of 7DOF manipulator.

This paper is arranged as follows. Section 2 briefly reviews RRT algorithm. Section 3 briefly introduces the Bi-directional RRT algorithm which is used in this paper. Section 4 presents a occlusion-free and collision-free constraints which is later integrated to the main algorithm. Section 5 explains the overall algorithm of a occlusion-free and collision-free path planning of a 7DOF manipulator with RRT. Section 6 presents a numerical simulation result of the proposed algorithm and the conclusion is drawn in Section 7.

II. Review of RRT

To generate a path planning algorithm of 7DOF manipulator of a humanoid robot, an RRT algorithm is is used in this paper. The main advantage of RRT algorithm compared with other sampling-based algorithm is its fast searching speed and convergence and retaining the advantage of other algorithm such as PRM as stated previously. In this section, some review of RRT algorithm is presented.

RRT was introduced by S. M. LaValle[9] as a randomized data structure which is designed to search a path planning. While RRT is originally designed to handle nonholonomic constraints, it is successfully used in holonomic, nonholonomic and kinodynamic problem with high degree of freedom. Some path planning problem which have a holonomic constraints is solved using RRT [10]. Moreover with a successful application of RRT, problems with complicated geometry was addressed using RRT[11][12]. Extension of RRT with kinodynamic problems are also existed in [13][14][15][16].

Single-RRT Planner was first introduced by S. M. LaValle[9]. However, inspired by classic bi-directional technique, Bidirectional-RRT Planner was introduced by S. M. LaValle[10]. Yershova et. al.[17] also introduced a dynamic-domain RRT to overcome the weakness of a classic RRT search method that cause them to explore slowly when the sampling domain is not well adapted to the problem.

III. Bidirectional RRT Algorithm

Bidirectional Rapidly-exploring Random Tree(BiRRT) algorithm is an extended version of RRT which is introduced by S. M. LaValle[] as mentioned previously. Bidirectional search method is constructed from a classical artificial intelligence method, making two random tree not just one. This algorithm is based on a greedy algorithm which connect the goal position to initial position resulting in two random trees. An BiRRT algorithm is as follows.

In algorithm 1, both two trees T_a and T_b are initialized.T_a is from the initial configuration and T_b is from the goal configuration(line 1). If the tree extension from T_a is successfully finished, tree extension from T_b starts toward the newly created nodes in T_a (line 3 and line 4). In algorithm 2, T_b is iteratively searching space toward the newly created node in T_a. Three results can be occur. One is Trapped condition, another is Advanced condition and the other is Reached condition. If the result of Extend(T,q) is Advanced condition, it tells that the generation of new node is succeeded and it can be extended again. If the function returns Trapped condition, it means that the generation is failed because of various reasons, for example, new node is outside search space limit. Finally, if the function returns Reached condition, the bridge node from T_a to T_b is generated and then then, generate a path from the initial configuration to the goal configuration.

Algorithm 1 BiRRT Planner(q_{init},q_{goal})

1: $T_a init(q_{init})$;$T_b init(q_{goal})$;
2: **for** k=1 to K do **do**
3:     $q_{rand} \leftarrow$ RandomConfig();
4:     **if** Extend(T_a,q_{rand}) $\neq$ Trapped **then**
5:         **if** Connect(T_a,q_{new}) = Reached **then**
6:             return Path(T_a,T_b);
7:         **end if**
8:     **end if**
9:     Swap(T_a,T_b);
10: **end for**
11: return Failure;

Algorithm 2 Connect(T,q)

S $\leftarrow$ Extend(T,q);
while S $\neq$ Advanced **do**
    S $\leftarrow$ Extend;
end while
return S;

IV. Occlusion-Free and Collision-Free Constraints Implementation

In this section the occlusion-free and collision-free constraints is discussed. Both constraints are important to path-planning the 7DOF manipulator. Occlusion-free means that the visibility of the vision sensor is not interfered by the movement of the manipulator and collision-free means that the manipulator parts does not collide with external obstacles when it moves along the path.

A. Occlusion-Free Constraint

The vision sensor can be monolithic camera or stereo camera which are used to gather visual information of the surrounding of a humanoid robot. The visibility of the vision sensor can be constructed by a pyramid shape. Fig. 1 illustrates this.

The red pyramid shape in Fig. 1 is the visibility of the vision sensor and the other parts are the left arm of the humanoid robot(7DOF manipulator). The occlusion-free

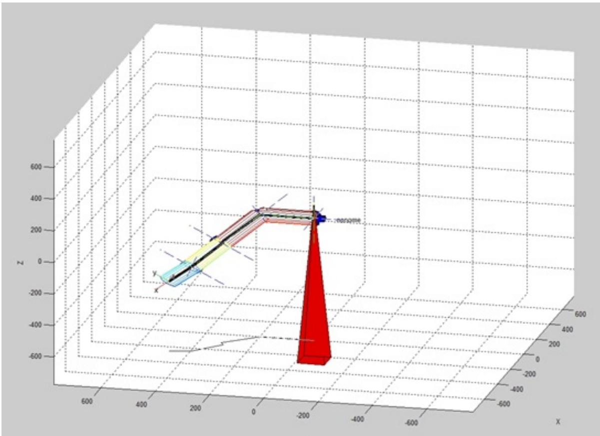

Fig. 1: The 7DOF manipulator(left arm of a humanoid robot) and visibility as a pyramid shape.

means that there are no intersections between the red pyramid and the left arm of a humanoid robot. Actually a function called $visibilitycheck()$ is constructed. The input to this function is the configuration of the left arm of a humanoid robot, the thickness of the arm, the position of the vision sensor(the vertex of the red pyramid), the direction cosine of the vision sensor(the orientation of the red pyramid), the height of the red pyramid, and the length of a segment of the base rectangle of the pyramid. With these input values, the $visibilitycheck()$ function calculates whether some intersection exists between the arm and the pyramid. This is done by representing each parts of the arm with three base vectors. The two of the three base vector is vectors comprising the two side line of the pyramid. And one vector is perpendicular unit vector of the two vectors. By representing each parts of the arm with these base vectors, the function can determine whether the parts of the arm have an intersection with the sides of the pyramid. If it has an intersection, the function returns true which means that the arm has actually interferes the visibility. it it does not have an intersection, the function returns false.

B. Collision-Free Constraint

V. Overall Procedure

The overall collision-free and occlusion-free path planning procedure for 7DOF manipulator is presented in this section. Fig. 2 illustrates this.

The simulation platform used in the simulation is Matlab and v-rep. Matlab is for simulating the main algorithm and v-rep is for sensing and actuating functionality of the humanoid robot. In Fig. 2, there are three main procedure. One is for extracting the obstacles and target object by segmentation and clustering. Before this procedure, RGBD sensor scanning is performed by the v-rep simulation engine. After determined

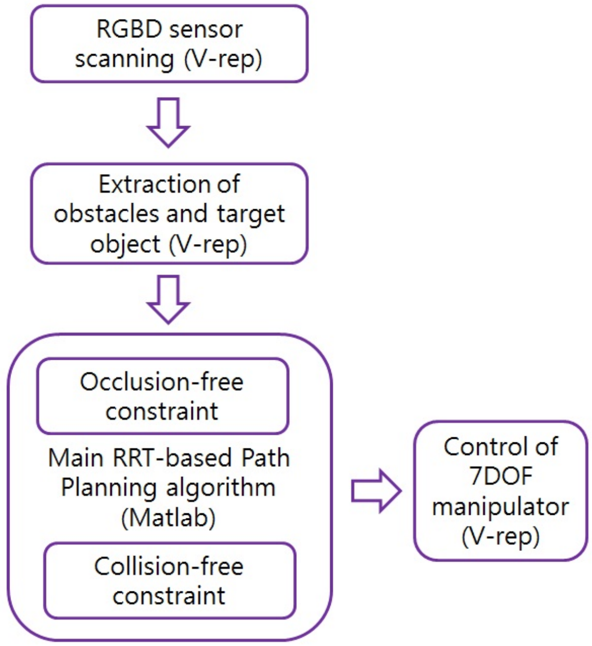

Fig. 2: Overall procedure.

the obstacles and the target object extraction, the 3D global coordinates and dimension of each obstacles and the target object is determined. With this information, the path-planning RRT algorithm with the occlusion-free and collision-free constraint is executed. This algorithm output the sequences of joint angle vector of 7DOF manipulator. Then the v-rep simulator receives this vector and actuates the 7DOF manipulator.

VI. The Simulation Result

In this section, the simulation result is presented. Fig. 3 and Fig. 4 represent the v-rep simulation scene. Fig. 3 represents the initial posture of 7DOF manipulator and Fig. 4 represents the final posture of 7DOF manipulator after performing a occlusion-free and collision-free path planning. In the scene, there are 2 sensors. One is RGBD sensor which scan the RGBD data. Its appearance is a small blue cylindrical form in the scene. The other is camera sensor which is used for visualizing the surrounding. Its appearance is a red colored motion-picture camera appearance in the scene. In the table, there are 3 object which is a sphere, a cuboid and a cylinder. The sphere is the target object and others are just obstacles. The small scene in the right upper corner is a scene from the camera. In Fig. 4, there are no occlusion and no collision with the obstacles after performing the path following.

VII. Conclusion

In this paper, a occlusion-free and collision-free path-planning of 7DOF manipulator using RRT is performed

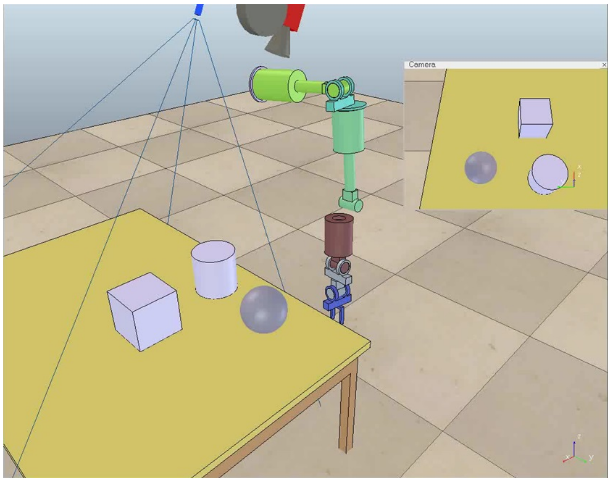

Fig. 3: v-rep scene(the initial posture of 7DOF manipulator).

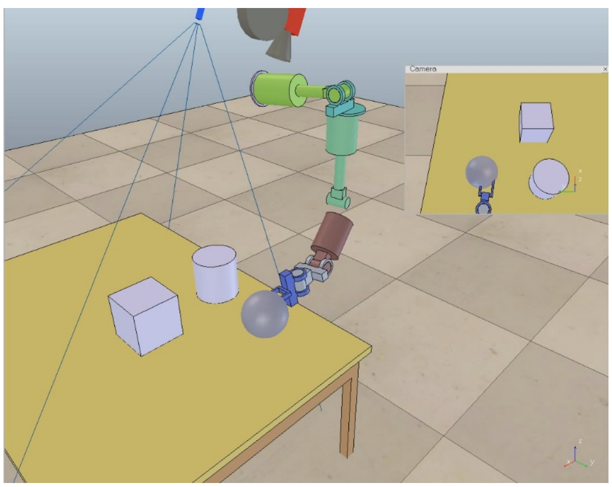

Fig. 4: v-rep scene(the final posture of 7DOF manipulator).

and verified by the numerical simulation. First, basic Bi-directional RRT algorithm which is used in the path-planning is discussed. And a occlusion-free constraints and collision-free constraints are explained. After explaining the overall procedure, we verified the path-planning algorithm with Matlab and v-rep simulator. Future work is to performing additional simulation with various objects and obstacles and upgrading the basic algorithm with RRT*.

references section

References

[1] Hirai, K. Hirose, M. Haikawa and T. Takaneka, "The development of Honda humanoid robot", *Proceedings of 1998 IEEE International Conference on Robotics and Automation*, pp. 1321-1326, May 1998.

[2] Jun-Ho Oh, David Hanson, Won-Sup Kim, Young Han, Jung-Yup Kim and Ill-Woo Park, "Design of Android Type Humanoid Robot Albert Hubo", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1428-1433, October 2006.

[3] F. Chaumette and S. Hutchinson, "Visual servo control part 1: Basic approaches", *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82-90, December 2006.

[4] F. Chaumette and S. Hutchinson, "Visual servo control part 2: Advenced approaches", *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 109-118, March 2007.

[5] A. Chan, E. A. Croft and J. J. Little, "Trajectory specification via sparse waypoints for eye-in-hand robots requiring continuous target visibility", *Proceedings of the 2008 IEEE International Conference on Robotics and Automation*, pp.2151-2156, September 2008.

[6] Y. Mezonaur and E. Chaumette, "Path planning for robust image-based control", *IEEE Transaction on Robotics and Automation*, vol. 18, no. 4, pp. 534-549, August 2002.

[7] Matthew A. Baumann, Donna C. Dupuis, Simmon Leonard, Elizabeth A. Croft and James J. Little, "Occlusion-Free Path Planning with a Probabilistic Roadmap", *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2151-2156, September 2008.

[8] Simon Leonard, Elizabeth A. Croft and James J. Little, "Planning Collision-Free and Occlusion-Free Paths for Industrial Manipulators with Eye-to-Hand Configuration", *The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5083-5088, October 2009.

[9] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path-planning", TR 98-11, Computer Science Dept., Iowa State University, October 1998.

[10] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approachto single query path planning", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 995-2000, April 2000.

[11] P. Choudhury and K. Lynch, "Trajectory planning for second-order underactuated mechanical systems in presence of obstacles", *Proceedings of the Workshop on Algorithmic Foundation of Robotics*, 2002.

[12] E. Ferre and J. P. Laumond, "An iterative diffusion method for part disassembly", *IEEE International Conference on Robots and Automation*, pp. 3149-3154 May 2004.

[13] M. S. Branicky and M. M. Curtiss,"Nonlinear and hybrid control via RRT", *Proceedings of International Symposium on Mathematical Theory of Netoworks and Systems*, June 2002.

[14] P. Cheng and S. M. LaValle, "Resolution complete rapidly-exploring random trees", *IEEE International Conference on Robotics and Automation*, pp.267-272, 2002.

[15] E. Frazzoli, M. A. Dahlet and E. Feron, "Real-time motion planning for agile autonomous vehicles", *AIAA Journal of Guidance and Control*, vol. 25, no. 1, pp. 116-129, 2002.

[16] J. Kim and J. P. Ostrowski, "Motion planning of aerial robot using rapidly-exploring random trees with dynamic constraints", *IEEE International Conference on Robotics and Automation*, pp. 2200-2205, September 2003.

[17] Anna Yershova, Leonard Jaillet, Thierry Simeon and S. M. LaValle, "Dynamic-Domain RRT: Efficient Exploration by Controlling the Sampling Domain", *Proceedings of IEEE International Conference on Robotics and Automation*, pp.3856-3861, April 2005.