October 2012

D* lite algorithm based path planning of mobile robot in static Environment

Pradipta kumar Das

Dhaneswar Rath Institute of Engineering and Management Studies Tangi, Cuttack, Orissa-754022, India,
daspradipta78@gmail.com

S .N. Patro

Dhaneswar Rath Institute of Engineering and Management Studies Tangi, Cuttack, Orissa-754022, India,
snpatro@gmail.com

C. N. Panda

Dhaneswar Rath Institute of Engineering and Management Studies Tangi, Cuttack, Orissa-754022, India,
cnpanda@ymail.com

Bunil Balabantaray

Synergy Institute of Technology, BBSR, Orissa-752101, India, bunil.balabantaray@gmail.com

Follow this and additional works at: <https://www.interscience.in/ijcct>

Recommended Citation

Das, Pradipta kumar; Patro, S .N.; Panda, C. N.; and Balabantaray, Bunil (2012) "D* lite algorithm based path planning of mobile robot in static Environment," *International Journal of Computer and Communication Technology*: Vol. 3 : Iss. 4 , Article 13.

DOI: 10.47893/IJCCT.2012.1158

Available at: <https://www.interscience.in/ijcct/vol3/iss4/13>

This Article is brought to you for free and open access by the Interscience Journals at Interscience Research Network. It has been accepted for inclusion in International Journal of Computer and Communication Technology by an authorized editor of Interscience Research Network. For more information, please contact sritampatnaik@gmail.com.

D* lite algorithm based path planning of mobile robot in static Environment

Pradipta kumar Das¹, Dr. S.N. Patro², C.N. Panda³, Bunil Balabantaray⁴

Dhaneswar Rath Institute of Engineering and Management Studies

Tangi, Cuttack, Orissa-754022, India

Synergy Institute of Technology, BBSR, Orissa-752101, India

¹daspradipta78@gmail.com, ²snpatro@gmail.com, ³cnpanda@ymail.com, ⁴bunil.balabantaray@gmail.com

Abstract— In this paper, we study the path planning for khepera II mobile robot in an unknown environment. The well known heuristic D lite algorithm is implemented to make the mobile robot navigate through static obstacles and find the shortest path from an initial position to a target position by avoiding the obstacles. and to perform efficient re-planning during exploration. The proposed path finding strategy is designed in a grid-map form of an unknown environment with static unknown obstacles. The robot moves within the unknown environment by sensing and avoiding the obstacles coming across its way towards the target. When the mission is executed, it is necessary to plan an optimal or feasible path for itself avoiding obstructions in its way and minimizing a cost such as time, energy, and distance. In our study we have considered the distance metric as the cost function*

Keywords— Robotics, Navigation, D lite algorithm, path planning.*

1. Introduction

Motion planning is one of the important tasks in intelligent control of a mobile robot which should be performed efficiently. It is often decomposed into path planning and trajectory planning. Path planning is to generate a collision free path in an environment with obstacles and optimize it with respect to some criterion [6, 9]. However, this environment may be imprecise, vast, dynamical and either partially or non-structured [4]. In such environment, path planning depends on the sensory information of the environment, which might be associated with imprecision and uncertainty. Thus, to have a suitable motion planning scheme in a cluttered environment, the controller of such kind of robots must have to be adaptive in nature. Trajectory planning is to schedule the movement of a mobile robot along the planned path. Several approaches have been proposed to address the problem of motion planning of a mobile robot. If the environment is a known static terrain and it generates a path in advance it said to be off-line algorithm. It is said to be on-line if it is capable of producing a new path in response to environmental changes.

Many studies have been carried out in path planning for different types of mobile robots. The works in [5] and [6] involve mapping, navigation, and planning

tasks for Khepera II mobile robot. In these works, the computationally intensive tasks, e.g., the planning and mapping tasks were not performed onboard Khepera. They were performed on a separate computer. The sensor readings and the motor commands are communicated between the robot and the computer via serial connection. The D* lite algorithm was used in these works for planning purposes. However the algorithm was applied only for an environment in which the locations of the obstacles are known in advance.

In our paper we proposed the use of D* lite algorithm for path planning in unknown environment unlike [5] and [6].

The rest of the paper is outlined as follows. Section 2 gives a brief of Khepera II components and its operation. The algorithm for the path planning is described in detail in section 3. Section 4 provides the result through snapshot taken from the implementation and section 5 concludes the paper.

2. KHEPERA II

Khepera II robot (figure a) is a miniature robot (diameter of 8cm) equipped with 8 built-in infrared range and light sensors, and 2 relatively accurate encoders for the two motors. The range sensors are positioned at fixed angles and have limited range detection capabilities. We numbered the sensors clockwise from the leftmost sensor to be sensor 0 to sensor 7 (figure b). Sensor values are numerical ranging from 0 (for distance > 5 cm) to 1023 (approximately 2 cm). The onboard Microprocessor has a flash memory size of 256KB, and the CPU of 8 MHz. Khepera can be used on a desk, connected to a workstation through a wired serial link. This configuration allows an optional experimental configuration with everything at hand: the robot, the environment and the host computer.

Figure 1(a): Khepera II

Figure 1(b): position of the sensors of Khepera II

3. Path planning algorithm

In this section the algorithm developed for the path planning of Khepera II mobile robot is presented. A grid map is build for the robot to navigate. In the grid map the robot can move in four neighbor directions vertically or horizontally. D* lite algorithm is used to computes or updates (when change in the environment is detected) the heuristic estimates of the distance corresponding to the shortest path from each state to the goal state. This is performed by searching backwards from the goal, i.e., propagating the distance estimation value to the successor nodes, starting from the goal node. The *D*Lite* path planner combines aspects of *A** search, the classic AI heuristic search method, and incremental search to plan near-optimal paths in partially known environments. Incremental search methods reuse information from previous searches to find solutions to series of similar tasks much faster than is possible by solving each search task from scratch. Each position has an associated cost function, $f(s) = g(s) + rhs(s) + h(s, s_{goal})$ where $g(n)$ is the generation cost or

movement cost from the starting position to any position in the grid, $rhs(n)$ is one step lookahead values based on the g -values and $h(s, s_{goal})$ is the estimated movement cost from a position to the target position. This is referred to as Heuristic cost. There are many different ways of defining the heuristic cost. In our implementation the cost $c(s, s_a)$ is the Euclidean distance between the next possible positions and the target position. The robot moves to the next position having minimum f value if there is no obstacle. If more than one neighbor has same value of $f(n)$ then the direction is chosen randomly. The algorithm is presented below.

$$rhs(s) = \begin{cases} 0 & \text{if } s = s_{start} \\ \min_{s_a \in Pred(s)} (g(s_a) + c(s_a, s)) & \text{otherwise.} \end{cases}$$

$$h(s, s_{goal}) = \begin{cases} 0 & \text{if } s = s_{goal} \\ c(s, s_a) + h(s_a, s_{goal}) & \text{otherwise.} \end{cases}$$

Procedure ComputeShortestPath ()

```
While (U.TopKey < Calculate Key (s_start) or rhs (s_start)
≠ g(s_start))
{
u ← U. Pop()
if (g(u) > rhs(u)) then
    g(u) ← rhs(u)
    for (s ∈ Pred(u))
    {
        Updatevertex(s)
    }
}
Else
g(u) ← ∞
For(s ∈ Pred (u) ∪ {u})
{
    UpdateVertex(s)
}
}
```

Procedure UpdateVertex (u)

```
{
If (u ≠ s_goal) then rhs ← min_{s ∈ Succ (u)} (c(u,s)+g(s) )
If (u ∈ U) then U.Remove (u)
If(g(u) ≠ rhs(u)) then U.Insert(u ,CalculateKey(u) )
}
```

Procedure CalculateKey(s)

```
{
return( min( g(s) , rhs(s) ) + h(s_start,s) , min ( g(s) ,
rhs(s) ) )
}
```

Procedure Initialize ()

```
{
```

```

U ← 0
  For(s ∈ S)
  {
    rhs ← ∞
    g(s) ← ∞
  }
rhs(sgoal) ← 0
U. Insert (sgoal, calculateKey (sgoal))
}
Procedure D*lite()
{
  Initialize ()
  ComputeShortestPath ()
  While (sstart ≠ sgoal)
  {
    sstart ← arg mins ∈ succ(sstart) ( c (sstart, s) + g(s) )
    Execute the instruction: move to sstart
    Scan the surrounding for obstacles
    If(obstacles)
    {
      Add two new states to both sides of the obstacle .
      Update the current map by linking the new states to
      the neighboring states in the current map with the
      appropriate behaviors and terminators
      Update the behavior and terminators of the
      Neighboring states to the new states
      UpdateVertex (previousState)
    }
  }
  ComputeShortestPath ()
}

```

Figure 2. Khepera network and its accessories

Figure 3. Environment setup for robot path planning

4. Experimental results

Extensive experiments were conducted to test the robot performance in path planning between any two positions in the unknown environment with various unexpected obstacles. Figure c and d shows the snapshots of the Khepera II mobile robot path planning in one such case study. Two static obstacles of different shape and size were considered for this case study.

(i)

(ii)

(iii) Figure d: Final position of the Robot

(iv)

Figure 4. Snapshots taken at different stages of path planning
(i) Initial movement of the robot
(ii) - (iii) Intermediate movement of the robot
(iv) Final position of the robot at the Goal

Figure 5. Robot path planning in a Grid map environment

5. Conclusion and Future works

In this paper, online path planning of mobile robot in unknown environment is presented using Khepera II mobile robot. The path planning is performed using Heuristic D* lite algorithm. With the help of extensive experiments the goal of this study is achieved. However only static obstacles were considered in the present study.

The result from this work provides a platform for developing robot control and also provides a useful tool for robotics education. For future work, we plan to include dynamic obstacles and to include the onboard camera in the study of online path planning of mobile robot.

References

- [1] A. Howard, M.J Matarić and G.S.Sukhatme, "An Incremental Self-Deployment Algorithm for mobile Sensor Networks, autonomous robots", special Issue on Intelligent Embedded Systems, 13(2), September 2002, pp. 113-126.
- [2] S. Florczyk, "Robot Vision Video-based Indoor Exploration with Autonomous and Mobile Robots ", WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim, 2005.
- [3] Danica Janglova, "Neural Networks in Mobile Robot Motion, pp. 15-22, International Journal of Advanced Robotic Systems, Volume 1 Number 1 (2004), ISSN 1729-8806.
- [4] Harlan, R. M, Levine, D. B., and McClarigan, S., "The Khepera robot and the kRobot class: a platform for introducing robotics in the undergraduate curriculum", proc. 32nd SIGCSE technical symposium on computer science education, pp. 105-109, 2001.
- [5] Harlan, R. M. and McClarigan, S., "Creating emergent behaviors: two robotics labs that combine reactive

- behaviors", *proc. 36th SIGCSE technical symposium on computer science education*. Feb. 2005, pp. 441- 445.
- [6] Murphy, R., "Introduction to AI Robotics", first edition, the MIT press, 2000.
- [7] <http://www.k-team.com/robots/khepera/index.html>.
- [8] Cen Li, Bryan Bodkin, James Lancaster "Programming Khepera II Robot for Autonomous Navigation and Exploration using the Hybrid Architecture ,ACMSE "09 March 19-21, 2009,Clemson ,Sc, USA
- [9] Stentz, A. "The focused D* algorithm for realtime replanning". In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 1652-1659, 1995.
- [10] D. Mackay DRDC Suffield "Path Planning with D*Lite" *Technical Memorandum DRDC Suffield TM 2005-242* December 2005
- [11] Amit Konar," *Artificial Intelligence and Soft Computing: Behavioral and Cognitive Modeling of the Human Brain*", 1st edition, CRC Press,1999.
- [12] I.K. Jung, K.B. Hong, S.K. Hong and S.C. Hong, "Path planning of mobile robot using neural network," in *Proc. IEEE Int. Symp. On Ind.Electron, Sch. Elect. Engg.*, 1999 *ISIE Bled Slovenia*, Vol. 3, 12-16 July,1999, pp. 979-983, DOI: 10.1109/ISIE,1999.796750.